AI Campus

EXPERIMENT

Programming Platform：DobotScratch
Hardware：Magician Lite

DOBOT

Shake Hands With The Future

CATALOGUE

Experiment 1

Repeated Stamping

Team Name: Team member: Date:

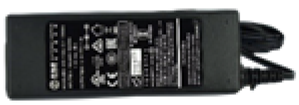
Overview

Stamps are one of teachers' common tools. Teachers often use them to evaluate students' homework or encourage their progress. Every time the students see the cute stamps, they feel happy and look forward to the next homework. However, sometimes there are too much homework to be assessed, manually stamping makes the teacher's wrist ache, and it becomes a very hard work. In this experiment, you will design an automatic stamping robot to solve the project problems from simple to complex. The steps are divided into three tasks: single stamping, multiple stamping, and infinite stamping.

Objectives

1. Understand the sequence structure by controlling the robotic arm movement instructions.
2. Understand the concept of coordinates.
3. Understand the finite loop by using commands of finite loops to control the robotic arm to stamp.
4. Understand the infinite loop by controlling the robotic arm to stamp with the infinite loop command.

Equipment

Icon	Name	Quantity
	Dobot Magician Lite robotic arm	1
	USB Type-C interface cable	1
	Suction cup	1
	Power Adapter	1
	Stamp	1
	Paperboard	Several

Experiment Requirements

- Take care when using electricity.
- Before the experiment, check whether the experiment equipment is complete and intact. If there is any omission or damage, please report to the teacher.
- Any specific operation in the experiment should be carried out according to the experiment manual. If you have any questions, please promptly ask the teacher.
- In the experiment, the joint of the robotic arm is in a working state after being powered on. Do not force it to move without pressing the unlock button.
- Equipment failure during the experiment must be reported to the instructor in time. Do not handle it by yourself.
- After the experiment, the equipment should be arranged well. You are allowed to leave the laboratory only after the inspection by the group leader.

Task 1: Single Stamping

A journey of a thousand miles begins with one step. The design of a stamping robot starts with making a single stamp. In this experiment, you will use paperboards to simulate the workbooks that the teacher needs to stamp, and then let the robotic arm complete the single stamping task.

1. Observations

First, the teacher will demonstrate a single stamping task. Pay attention to the movement of the robot arm, and then complete the steps in the blank.

Step 1: Choose the proper end ________. (Hint: fill in the suction cup or gripper)

Step 2: Move to the stamp position, __________ the stamp.

Step 3: Move to ________ position， and stamp.

2. Steps

(1) Prepare Hardware

Step 1: Design the placement of paperboards, stamps, and robotic arms, and place them according to the design diagram 1.1.

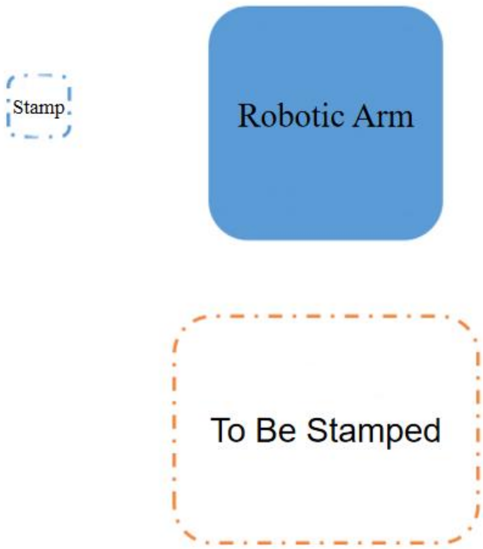


Figure 1.1 Schematic diagram of placement

Step 2: Install the suction cup and power on the Magician Lite, as shown in Figure 1.2.

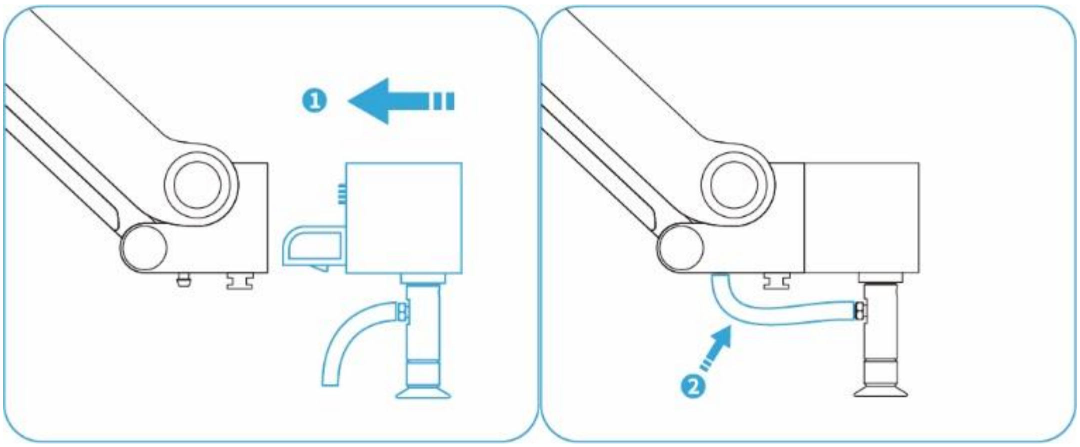


Figure 1.2 Installing the suction cup

Step 3: Power on the computer.

Step 4: Use a Type-C interface cable to connect Magician Lite and the computer, as shown in Figure 1.3.

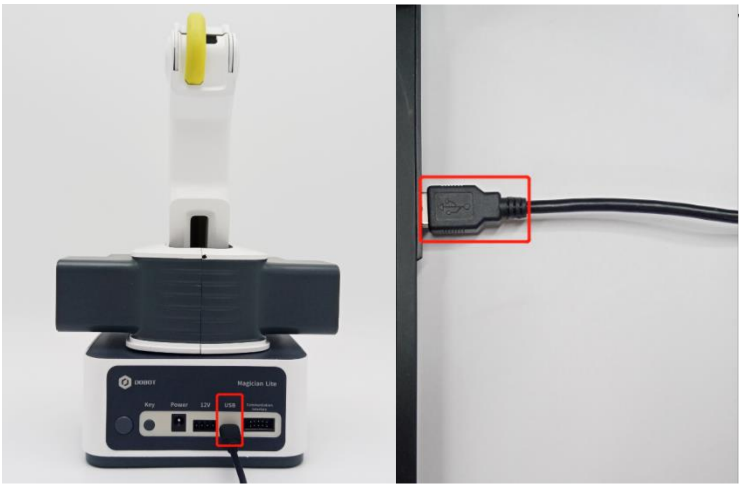


Figure 1.3 Connecting with Type-C data cable

(2) Design Program

Step 1: According to your observation, read the process flowchart of one single stamping, as shown in Figure 1.4.

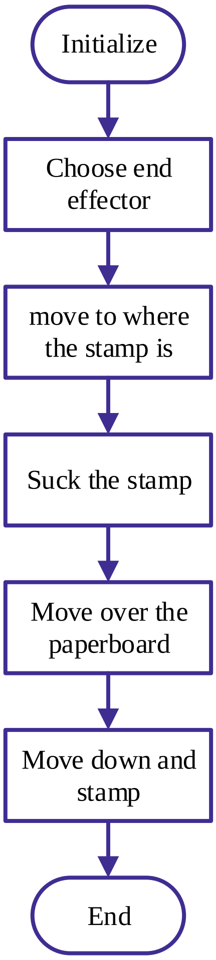


Figure 1.4 Flowchart of single stamping

Step 2: Power on the computer, move the mouse pointer to the DobotScratch software icon and double click it to start the DobotScratch software, as shown in Figure 1.5 and 1.6.

Figure 1.5 DobotScratch software icon

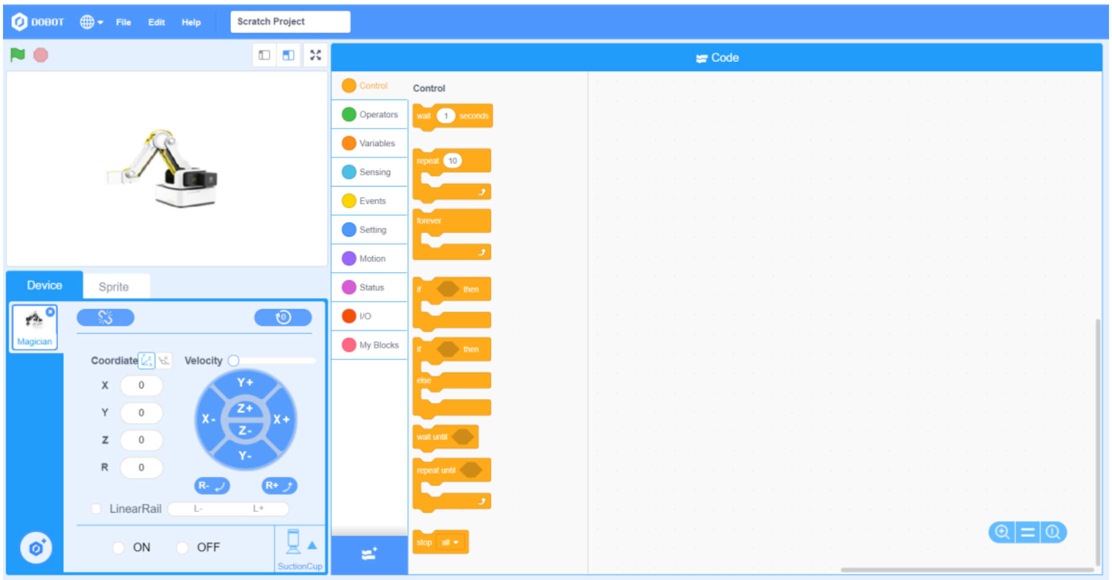


Figure 1.6 Interface of the software

Step 3: Click with the left mouse button and choose the device Magician Lite, as shown in Figure 1.7.

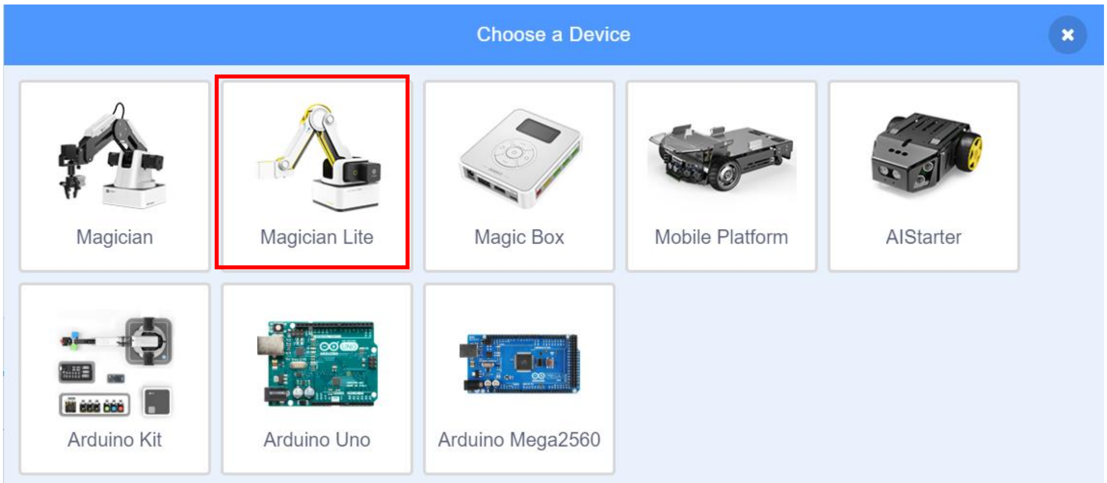


Figure 1.7 Choosing device

Step 4: Click with the left mouse button to display the device connection interface. Click[redacted] Connect with the left mouse button to connect the device with Magician Lite, as shown in Figure 1.8.

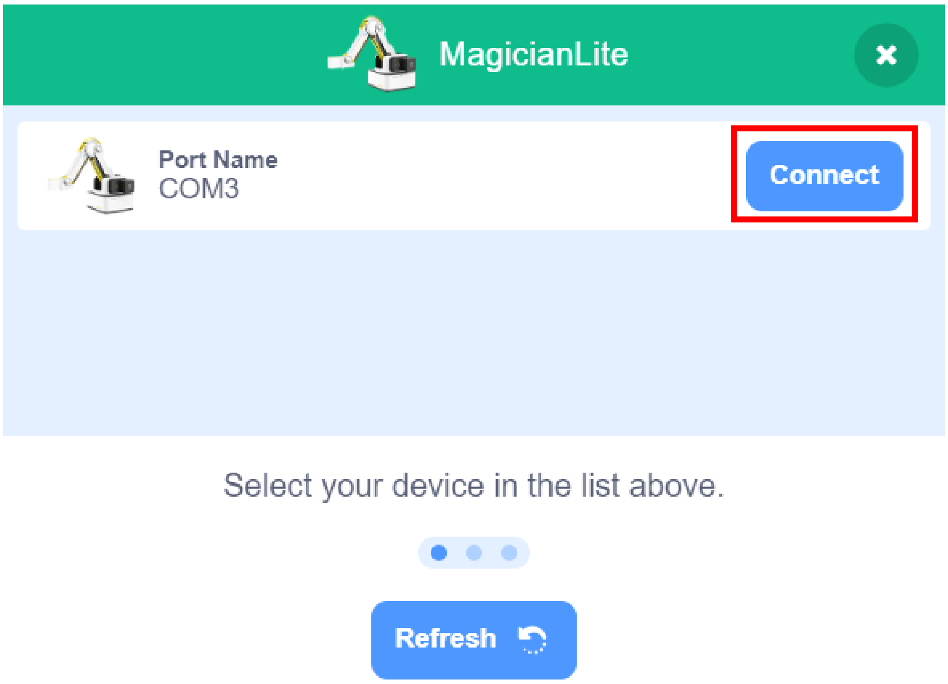


Figure 1.8 Device connection

Step 5: To determine whether the device is connected successfully, check the control panel for the coordinates of the robotic arm end, as shown in Figure 1.9. The picture on the left indicates a successful connection, and the picture on the right a connection failure.

Figure 1.9 Determining whether the connection is successful

☛ After the connection is successful, try to control the robotic arm with the control panel.

Step 6: Set the triggering method of the program. Click the "**Events**" tab, and drag the "When the green flag is clicked" block, as shown in Figure 1.10.

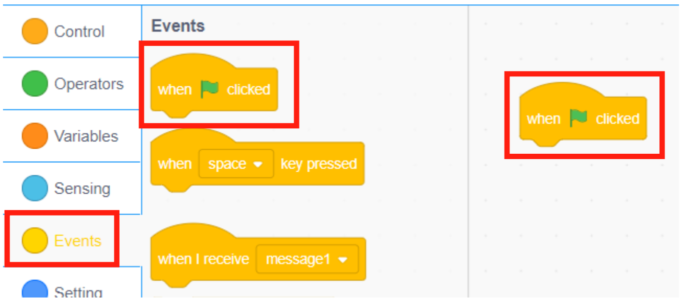


Figure 1.10 "When the green flag is clicked"

Step 7: Set the end tool. Click on the "**Setting**" tab, drag the "End Tool setting (gripper or suction cup)" block, and put it together with the previous block, as shown in Figure 1.11.

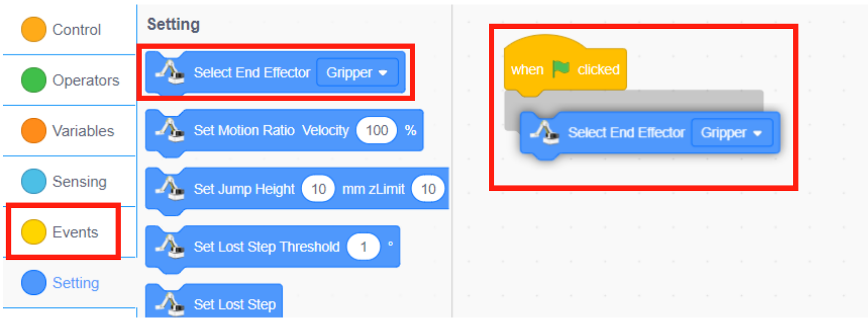


Figure 1.11 Setting the end tool

Step 8: Press the unlock button on Magician Lite and move the end of the robotic arm to the stamp. Click the "**Motion**" tab, drag the "Jump To (X, Y, Z, R)" block, and put it together with the previous block. Right-click and select "Fill coordinates", as shown in Figure 1.12.

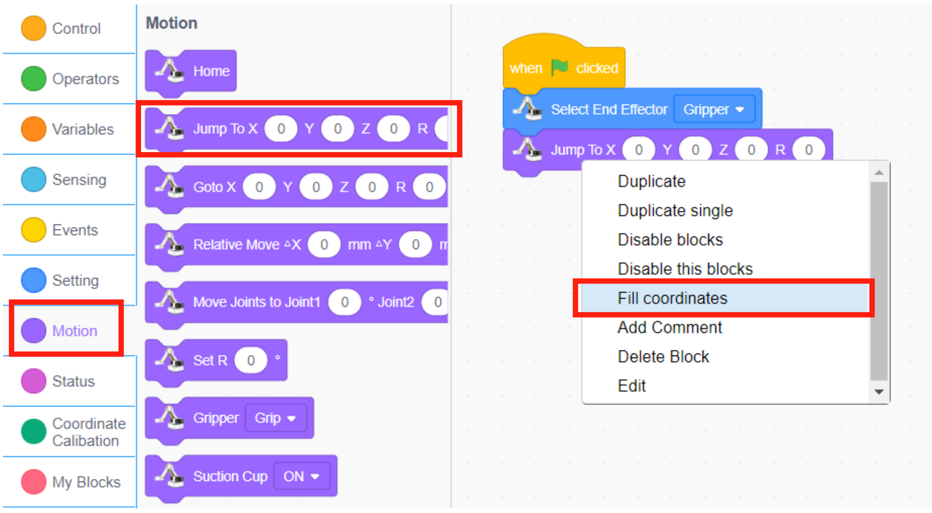


Figure 1.12 JUMP move onto the stamp

Coordinates are used to indicate the position of a point, which is represented by an array. For example, a point on a plane is represented by (X, Y), and a point in the space is represented by (X, Y, Z).

Step 9: Pick up the stamp. Click on the "**Motion**" tab, drag the "Suction Cup" block, and concatenate it with the previous block, as shown in Figure 1.13.

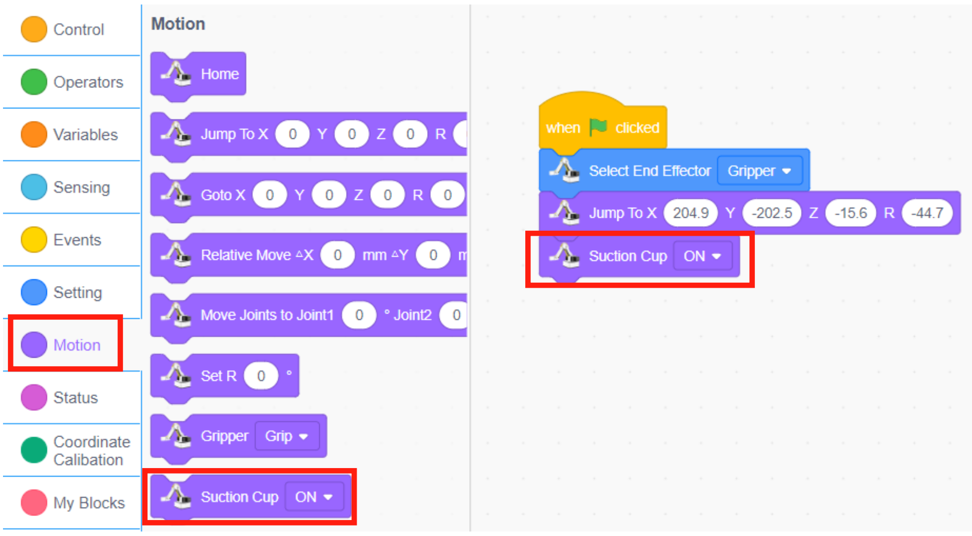


Figure 1.13 Suction of the stamp

Summarize the method for the robotic arm to read the position.

Step 10: Move the end of the robotic arm over the paperboards. Click the "**Motion**" tab, drag the "Jump To (X, Y, Z, R)" block and put it together with the previous block. Right-click and select "Fill-in Coordinates", as shown in Figure 1.14.

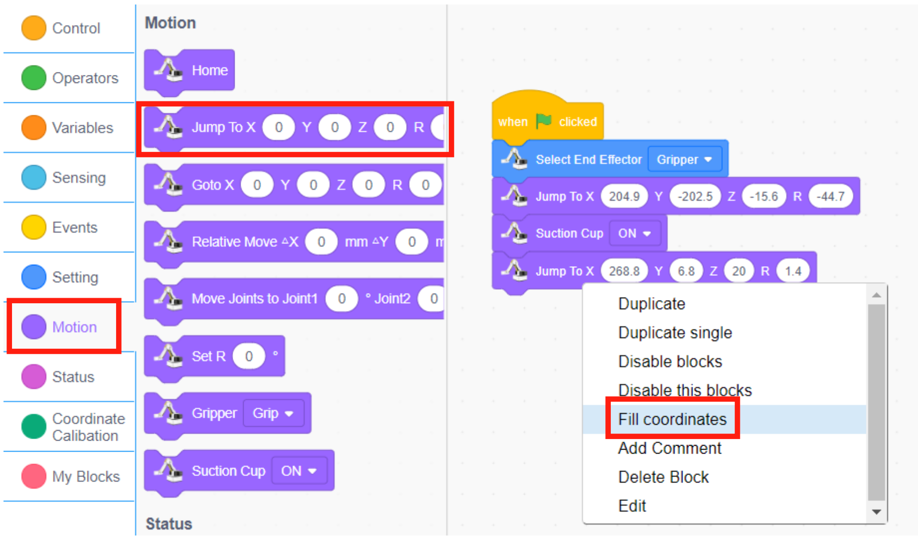


Figure 1.14 JUMP move onto the paperboards

To stamp, you need to move the end of the arm over the paperboards and push down The Z-axis coordinates obtained in step 11 need an addition. For example: The Z-axis coordinate obtained is -30 and the descending height is set to 50; then the Z-axis coordinate above the paperboards is (-30 + 50 = 20).

Step 11: Click the "**Motion**" tab, drag the "Relative Move (X, Y, Z, R)" block and concatenate with the previous block. In the "relative motion (X, Y, Z, R)" block, the Z-axis coordinate is the decreasing height of 50, so Z is " -50 " because the direction is downward, as shown in Figure 1.15.

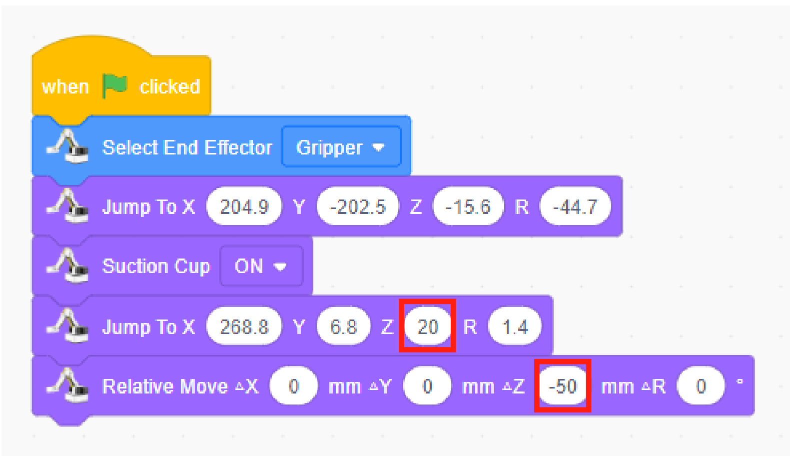


Figure 1.15 Position above the paperboards and the descending distance

What are the instructions to control the movement of the robotic arm?

What are the instructions to control the suction cup?

According to the problem-solving steps, the program structure that is executed sequentially from top to bottom is called **sequence structure**. Sequence structure is the simplest and most commonly used structure in a program.

3. Summary

(1) Methods for obtaining the position of the robotic arm: ___________

____

(2) The instructions to control the movement of the robotic arm are: _

____

(3) The instructions to control the suction cup are: _______________

____

(4) The characteristics of the sequence structure in the program are: _

____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) the unfinished ones.

Assessment Content	Completion Status
I've known how to get the position of the robotic arm	
I've used the instructions to make a JUMP movement	
I've used the instructions to control the suction cup	
I've known the sequence structure	
I've completed the single-stamping program	

Task 2: Multiple Stamping

Sometimes a teacher has a lot of workbooks to stamp and evaluate. The automatic stamping robot needs to finish the task of automatic stamping according to the number of workbooks set by the teacher. In this experiment, you will complete the task of automatic stamping by the robotic arm for finite times.

1. Observations

The teacher will demonstrate the multiple stamping task. Observe the movement of the robotic arm and fill in the blank lines in the steps below.

Step 1: Choose the proper end ________. (Hint: fill in the suction cup or gripper)

Step 2: Move to the stamp position, ________ the stamp.

Step 3: Move to ________ position, and stamp.

Step 4: ________ step 2 to step 3.

2. Steps

(1) Prepare Hardware

The hardware setup for task 2 is the same as that for task 1. Refer to task 1 to set up the hardware environment for task 2.

(2) Design Program

Step 1: Compare to the single-stamping flowchart of the robotic arm. According to the observations, read the 10-time robotic arm stamping flowchart, as shown in Figure 1.16.

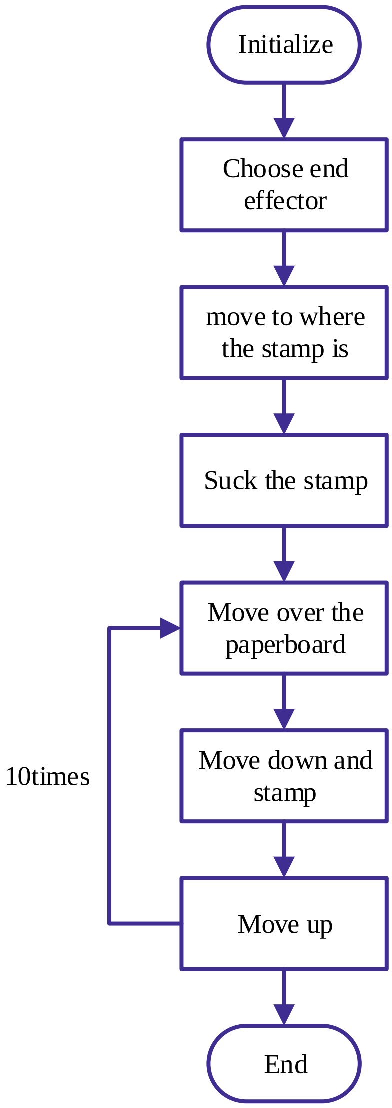


Figure 1.16 10-time robotic arm stamping flowchart

Step 2: Review how to write a single-stamping program, as shown in Figure 1.17.

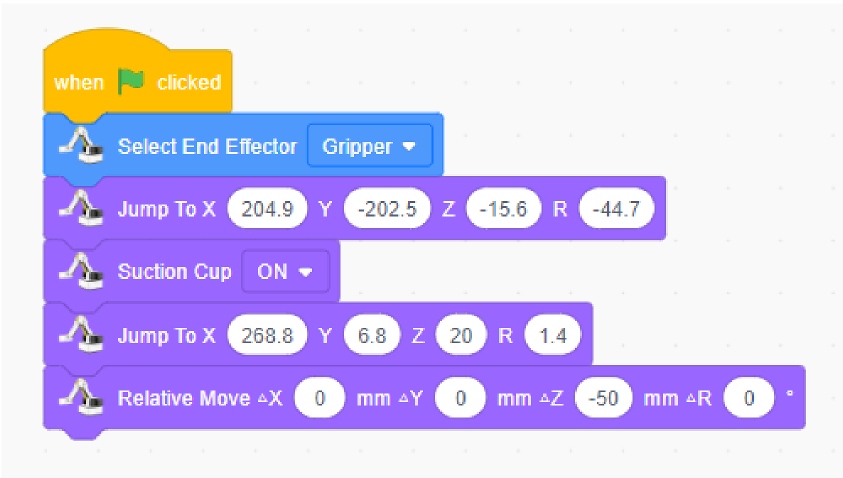


Figure 1.17 Single-stamping program

Review task 1: Single-stamping program. Summarize the difference between the single-stamping program and the multi-stamping program.

Step 3: Move the robotic arm upward, click on the "Motion" tab, drag out the "Relative Move (X, Y, Z, R)" block, and set the Z-axis coordinate relative movement upward by 50, as shown in Figure 1.18.

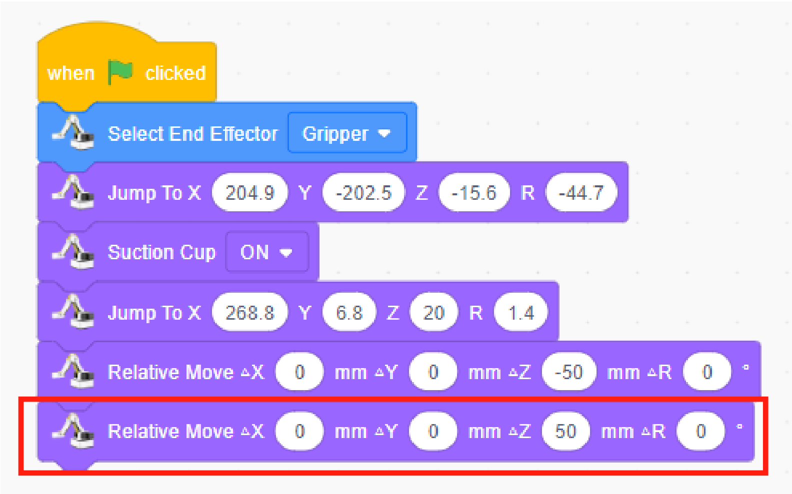


Figure 1.18 Moving the robotic arm upward

Step 4: Repeat the stamping 10 times. Click the "**Control**" tab, drag out the "repeat (10)" block, and repeat the ink dipping and stamping actions, as shown in Figure 1.19.

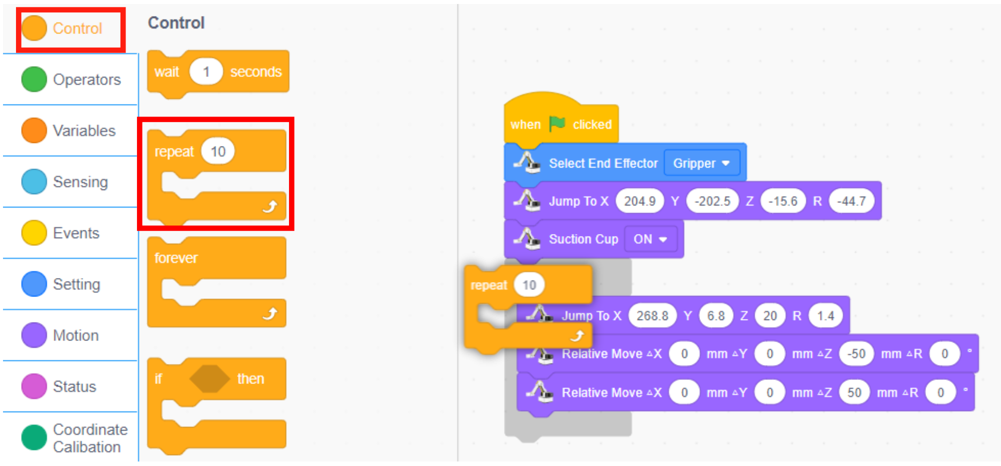


Figure 1.19 Repeating 10 times

A structure in a program that requires an action to be performed repeatedly is called a loop structure.

Step 5: Integrate the stamping process for 10 times, as shown in Figure 1.20.

Figure 1.20 10-time stamping program

Try to write the 20-time stamping program by yourself.

3. Summary

(1) The instructions for a finite number of loops are: ____________

____

(2) Similarities between the single stamping task and the finite stamping task: ____

____

(3) The characteristics of the loop structure are: _______________

____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) the unfinished ones.

Assessment Content	Completion Status
I've used a finite loop instructions	
I've known the loop structure	
I've completed the finite stamping program	

Task 3: Infinite Stamping

When a school holds a large-scale event, many certificates need to be prepared in advance to honor the outstanding students. When there are too many certificates, it is unrealistic to set the number of stamps for the robot. This experiment attempts to design an infinite stamping robot to assist the teacher in completing the stamping task. Here, we use paperboards to replace the certificates.

1. Observations

The teacher will demonstrate the infinite stamping task. Observe the movement of the robotic arm and fill in the blank lines in the steps below

Step 1: Choose the proper end ________. (Hint: fill in the suction cup or gripper)

Step 2: Move to the stamp position, __________ the stamp.

Step 3: Move to ________ position, and stamp.

Step 4: Move to ________ position, and place the stamp.

Step 5: Move to ________ position, and suck the stamped paperboards.

Step 6: Move to ________ position, and place the paperboards.

(Hint: fill in the position of the robotic arm movement)

Step 7: ________ step 2 to step 6.

2. Steps

(1) Prepare Hardware

The hardware setup for task 3 is the same as that for task 1. Refer to task 1 to set up the hardware environment for task 3.

(2) Design Program

Step 1: Unlike the multi-stamping flow by the robotic arm, the infinite-stamping flow also needs to solve the problem of replacing the stamped certificate. According to your observation, read the flowchart of completing stamping for infinite times, as shown in the Figure 1.21.

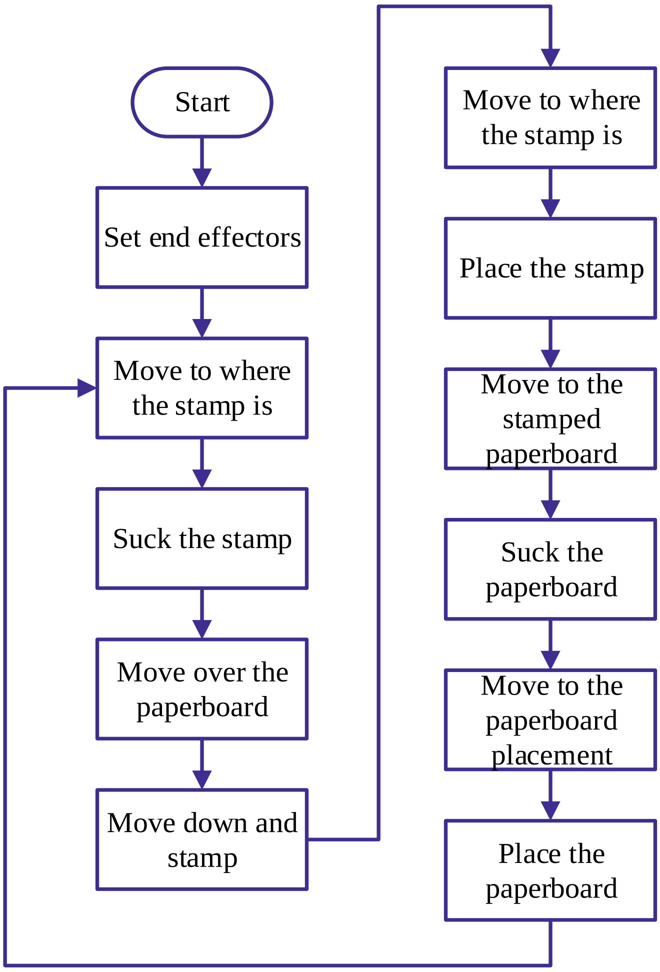


Figure 1.21 Infinite-stamping flowchart

Think about it: What problem is solved when the stamp is released after stamping?

Step 2: Write a single-stamping program.

Step 3: Return to the position of the stamp. Move the mouse pointer to the "Jump To the stamp" block, right-click it, and select "copy single", put the copied block to the bottom of the program, and enable the end of the robotic arm to return to the position of the stamp, as shown in Figure 1.22

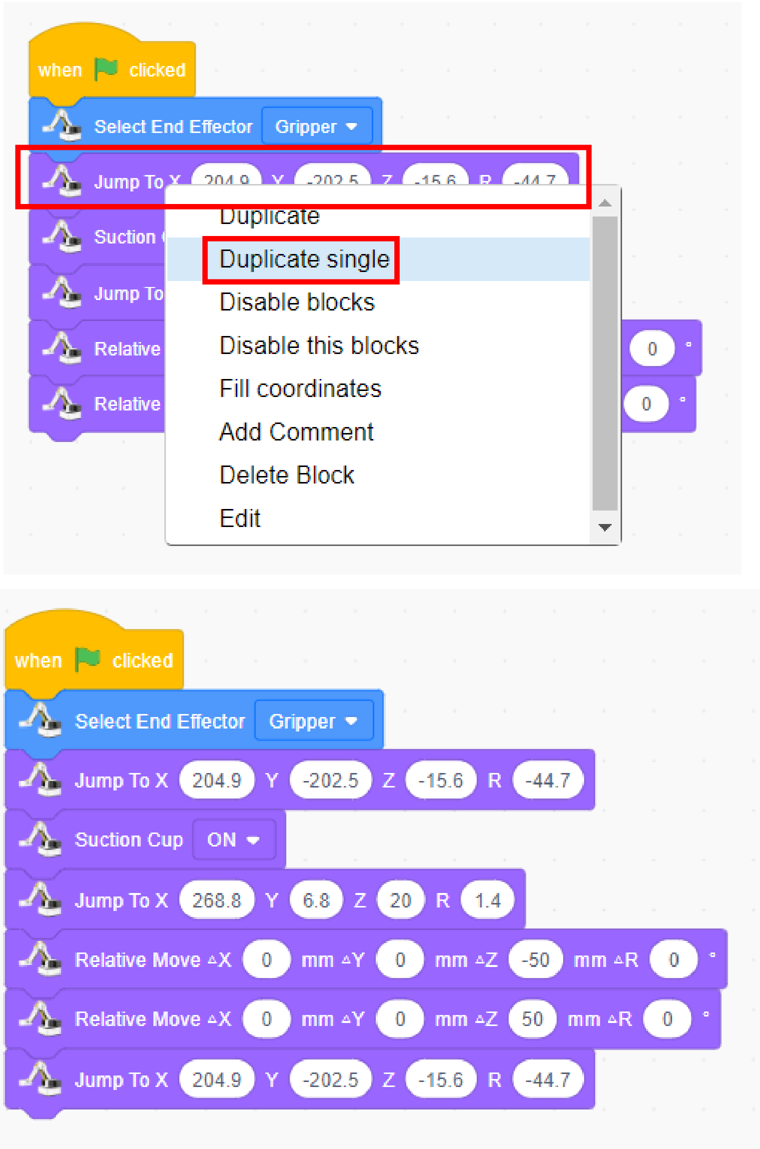


Figure 1.22 Copy "Jump To the stamp"

Step 4: Release the stamp. Click the "**Motion**" tab, drag the "Suction Cup" block, click the drop-down triangle and choose "Release", as shown in Figure 1.23.

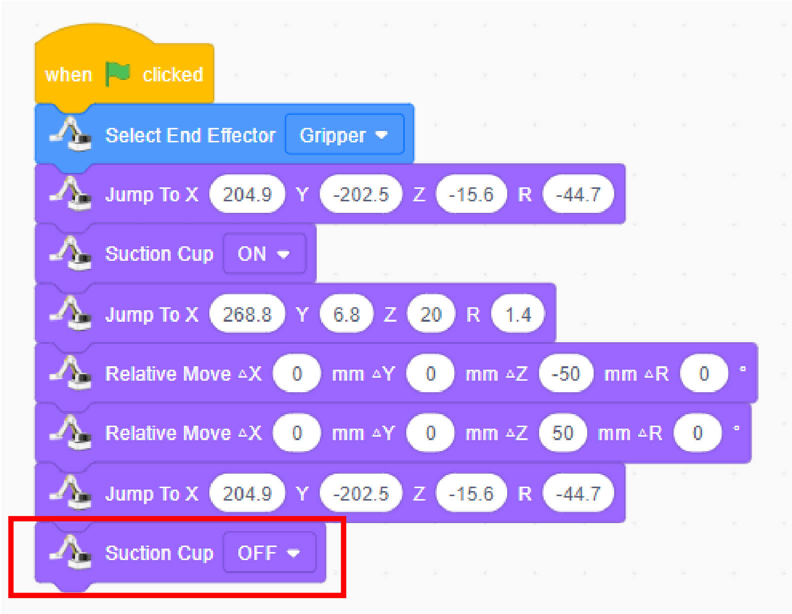


Figure 1.23 Releasing the stamp

Step 5: Suck the certificate. Move the robotic arm to the certificate, and enable the suction cup to "suck", as shown in Figure 1.24.

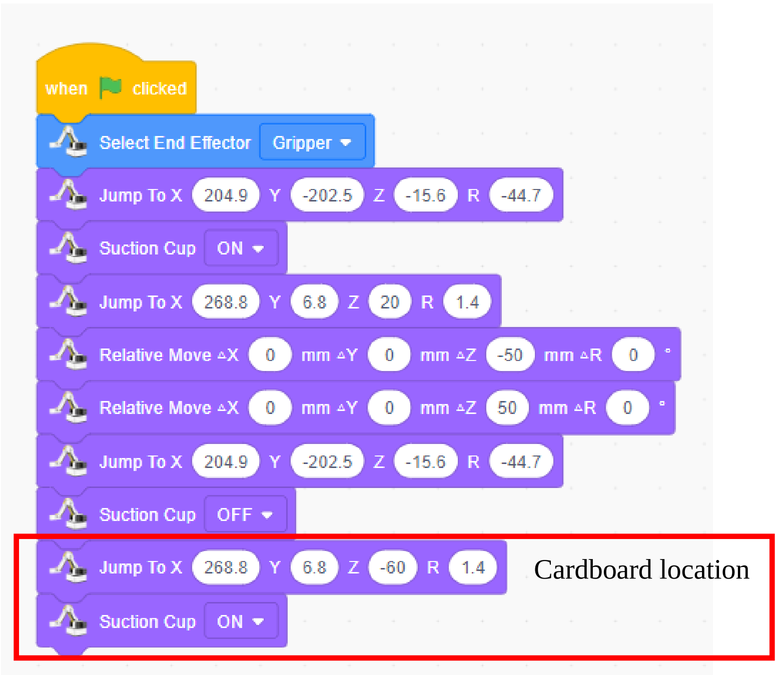


Figure 1.24 Sucking the paperboards

Why are the Z-axis coordinates of the paperboard's position different from those of the certificate's position in step 5?

Step 6: Place the paperboards. Move the robotic arm to the placement area and "release" the suction cup, as shown in Figure 1.25.

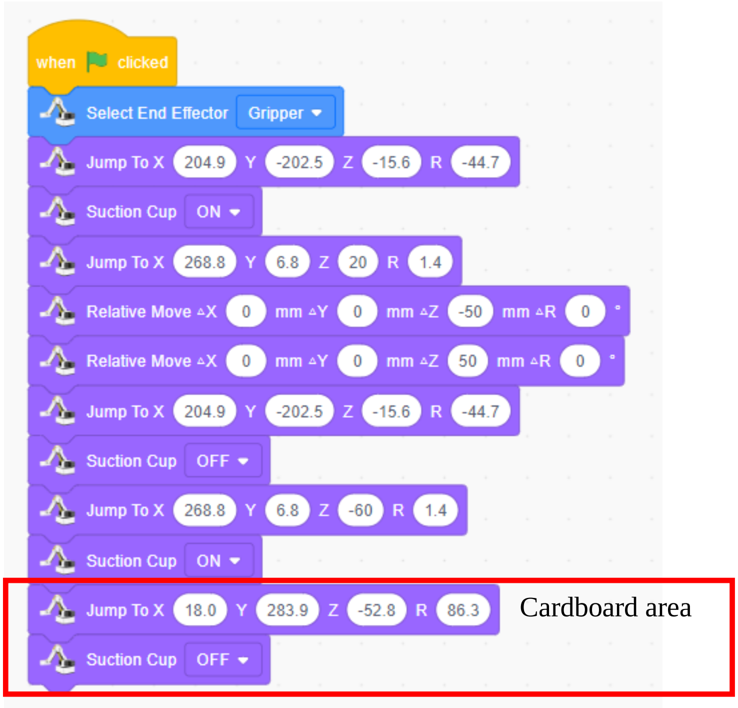


Figure 1.25 Releasing the certificate

Step 7: Wait for the ink to dry. Click the "**Control**" tab and drag the "wait (1) second" block to put it together with the previous block, as shown in Figure 1.26.

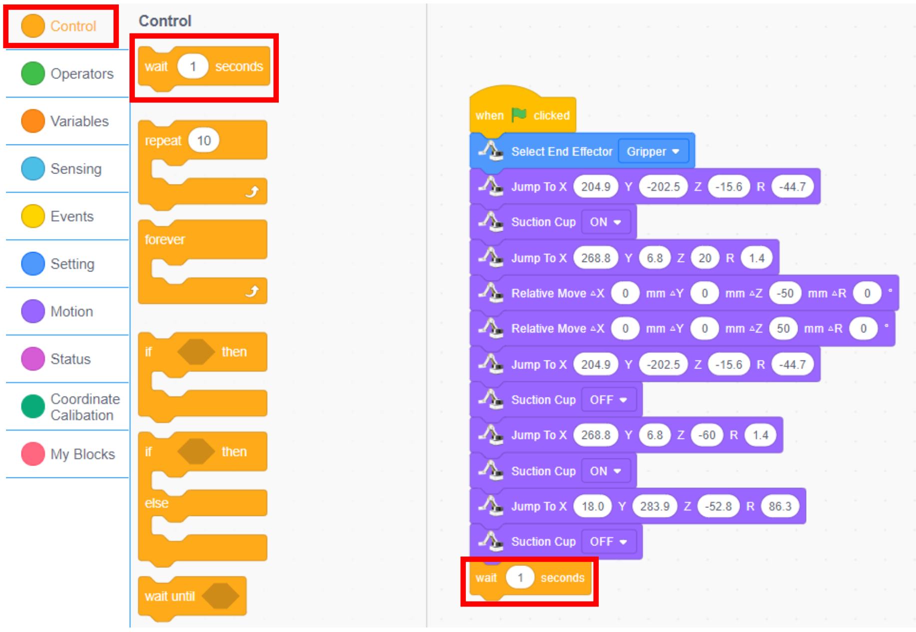


Figure 1.26 Waiting for the ink to dry

Step 8: Repeat the stamping action. Repeat the action of dipping the ink - stamping - releasing the stamp - sucking the certificate - releasing the certificate. Click the "**Control**" tab and drag out the "forever" block, as shown in Figure 1.27.

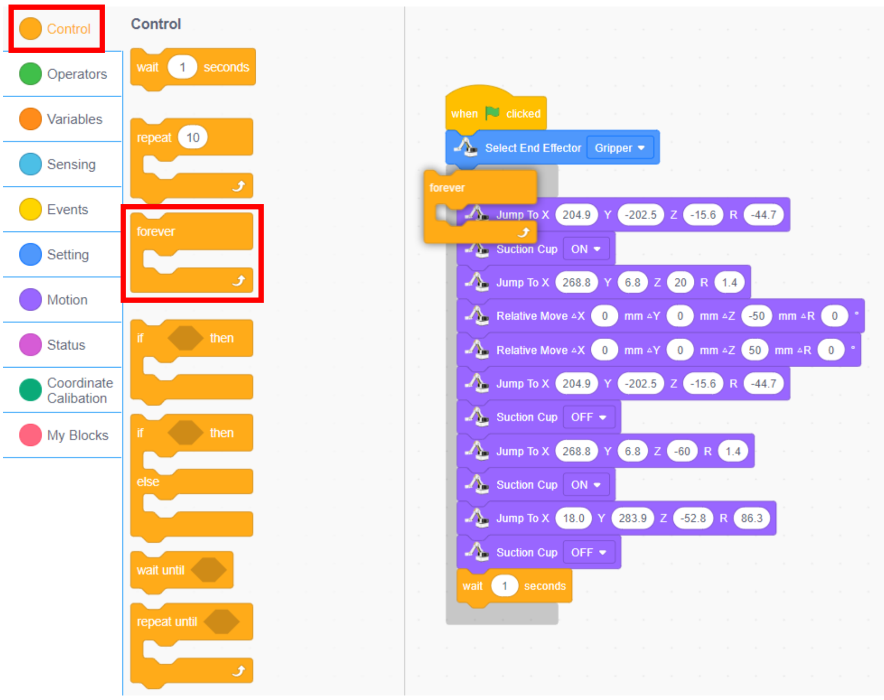


Figure 1.27 Repeating stamping action

Step 9: Integrate the program, as shown in Figure 1.28.

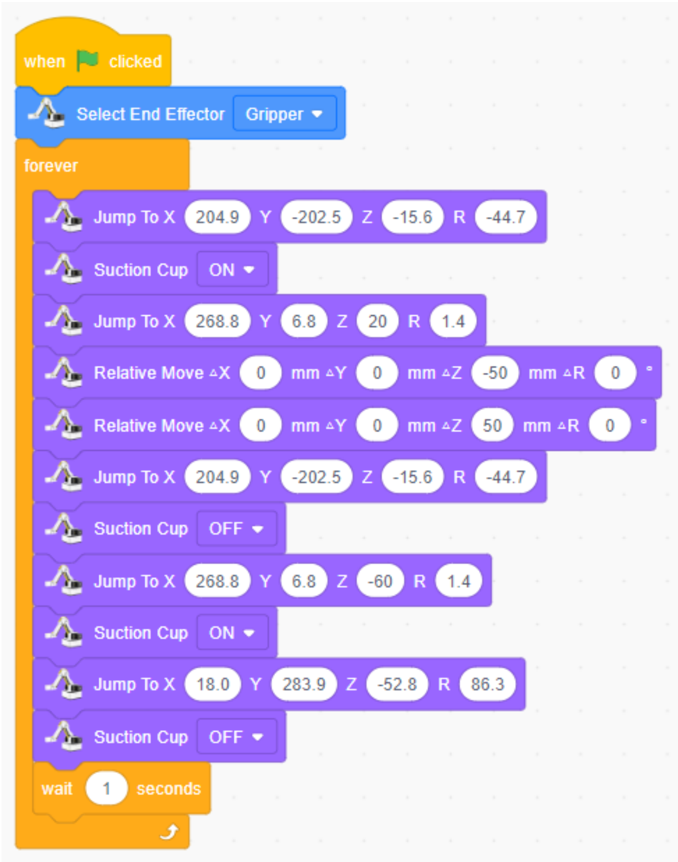


Figure 1.28 Infinite-time stamping

3. Summary

(1) The instructions for the infinite loop are:____________________

____

(2) The difference between an infinite loop and a finite loop is:____

____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) the unfinished ones.

Assessment Content	Completion Status
I've used the infinite loop instruction	

I’ve solved the problem that requires manpower to manually take away the stamped certificate	
I’ve completed the infinite-stamping program	

5. After School Extension

Infinite-stamping program can free teachers from monotonous and repetitive work. After the robot keeps stamping for a while, the paperboards are dwindling and becoming too thin for the stamp to touch. How do we solve this problem?

Experiment 2

Campus Porter

Team Name: Team member: Date:

Overview

Human power is limited. The weight that can be carried by a single person is also limited. In addition, repeated transportation causes fatigue and reduce work efficiency. In schools, boxes of books and laboratory supplies often need to be moved to designated locations. Using robots for carrying can not only save manpower, but also improve efficiency. This experiment is designed to help school staff improve efficiency. The solution is provided from simple to complex, including repeating the same action, repeating forward movement, carrying two objects, and carrying four objects.

Objectives

- By studying variable-related blocks in Scratch, understand the role of variables and how to use them.
- By calculating the height change of objects, understand and derive the formula of height changing.
- By controlling the robotic arm to repeatedly execute the same action, master the block of repeat execution.
- By controlling the robotic arm to repeatedly execute the same action, learn about the block of repeated execution with conditional judgment.

Equipment

Icon	Name	Quantity
	Dobot Magician Lite robotic arm	1
	Suction cup	1
	Power adapter	1
	USB type-C interface cable	1
	Objects	4

Experiment Requirements

- Take care when using electricity.
- Before the experiment, check whether the experiment equipment is complete and intact. If there is any omission or damage, please report to the teacher.
- Any specific operation in the experiment should be carried out according to the experiment manual. If you have any questions, please promptly ask the teacher.
- After the robotic arm is powered on, the motor of the robotic arm is in a working state. Do not force it to move without pressing the unlock button.
- After the experiment, the equipment should be arranged well. You are allowed to leave the laboratory only after the inspection by the group leader.

Task 1: Repeat the Same Action

Through the study of Experiment 1, you are able to control the robotic arm and make it perform related operations. Now, you will use the "Repeat execution" block in Scratch programming to make the robotic arm repeat the upward movement.

1. Observations

First, observe how the teacher make the robotic arm repeatedly perform the same action. Then, fill in the blank in the table below according to your observation.

Number of repeated upward movements by the robotic arm: the distance by which the robot arm movies upward each time: (Fill in: same or different).

2. Steps

(1) Prepare Hardware

Step 1: Connect Magician Lite to a power supply.

Step 2: Use a USB cable to connect Magician Lite to the computer and turn on Magician Lite.

(2) Design Program

Step 1: According to your observation, read the flowchart of moving upward three times by the robotic arm, as shown in Figure 2.1.

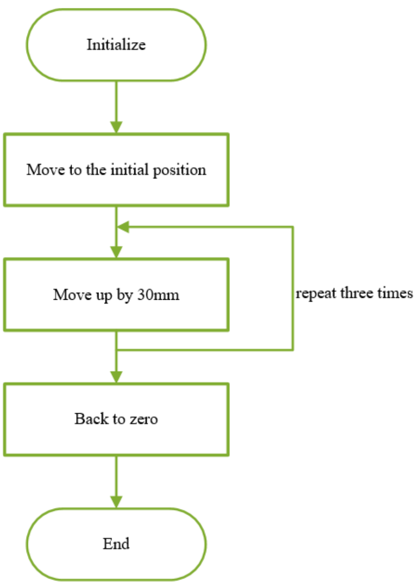


Figure 2.1 Flowchart of repeating upward movement

Step 2: Add the device and connection in the software interface. Add Magician Lite to the device list area at the bottom left of the DobotScratch interface, and click the connection button in the control area to connect to Magician Lite, as shown in Figure 2.2.

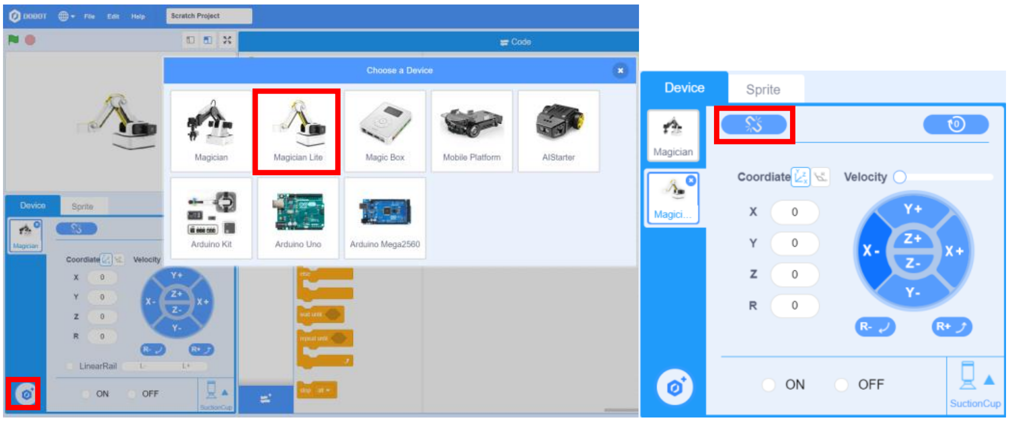
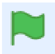


Figure 2.2 Adding and connecting to the device

Step 3: Trigger the program. In the Scratch interface, click the "**Events**" tab and drag the "when  is clicked" block to the code area, as shown in Figure 2.3.

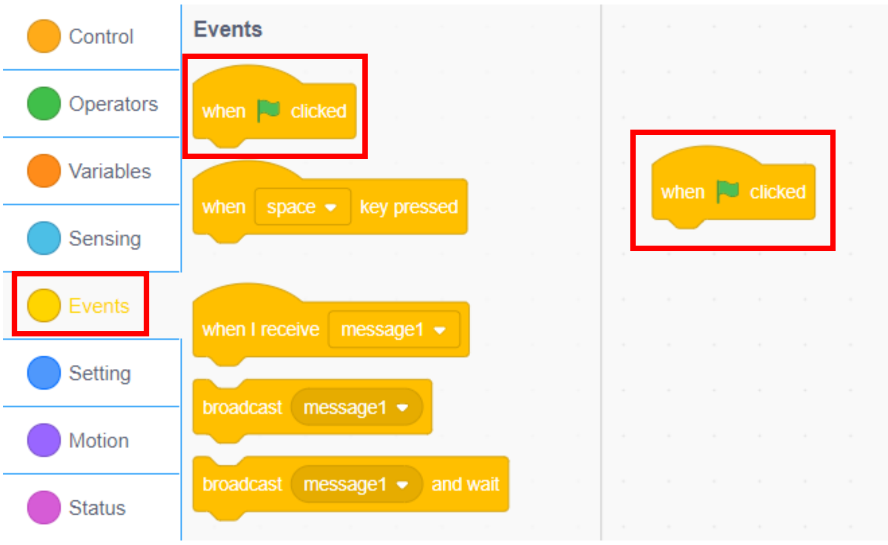


Figure 2.3 Starting block concatenation

Step 4: Move to the initial position. In the Scratch interface, click the "**Motion**" tab, drag the "Motion" block to the code area, set the X, Y, and Z values to 220, 0, and -30, and set the motion type to "Straight Line" motion, as shown in Figure 2.4.

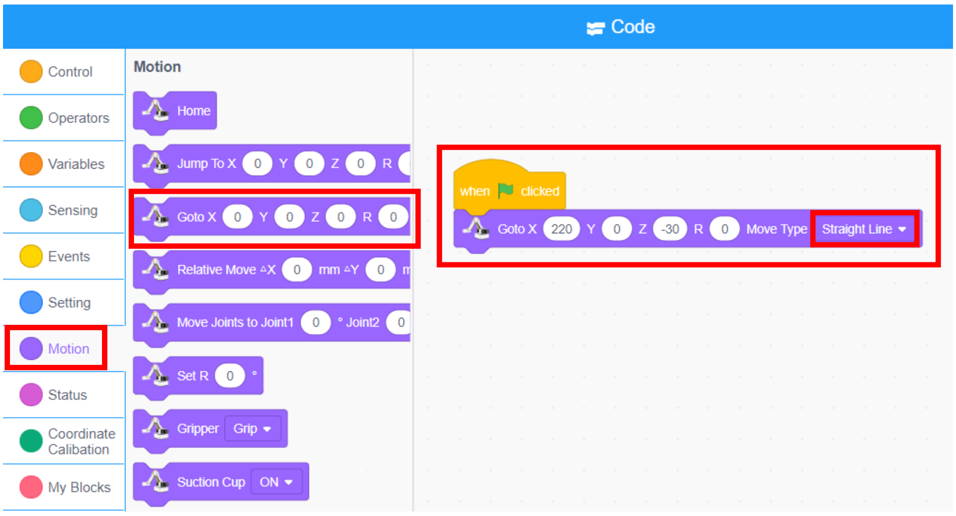


Figure 2.4 Motion block

Step 5: Create a variable. In Scratch, click the "**Variables**" tab, click "Make a Variable", name the new variable "i" in the pop-up interface, select applying for all sprites, and click OK. During the repeated execution, the Z-axis coordinate value increases as the variable increases. Figure 2.5 shows how the variable is created.

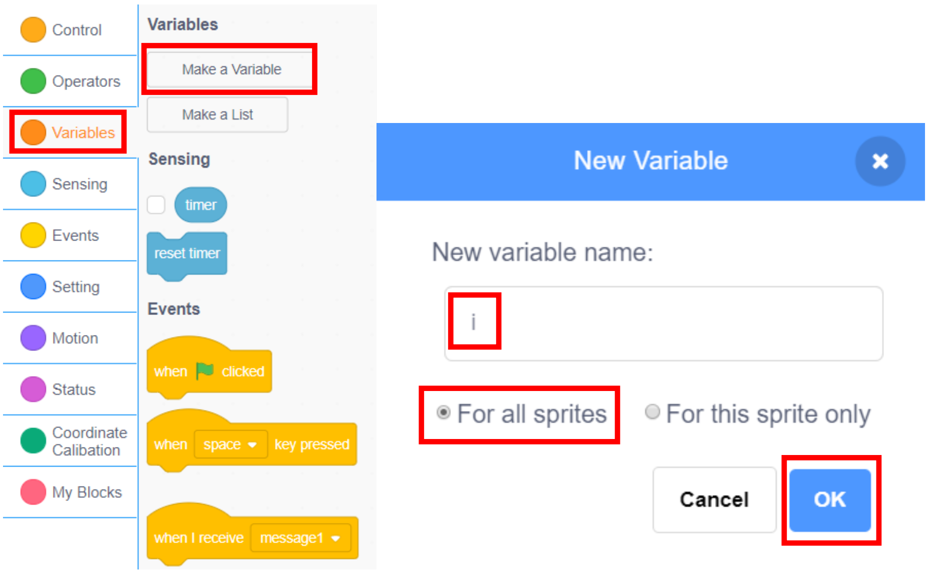


Figure 2.5 Creating a variable

Step 6: Set the variable. Click the "**Variables**" tab, drag the "Set i to 0" block to the code area, and put it together with the previous block, as shown in Figure 2.6.

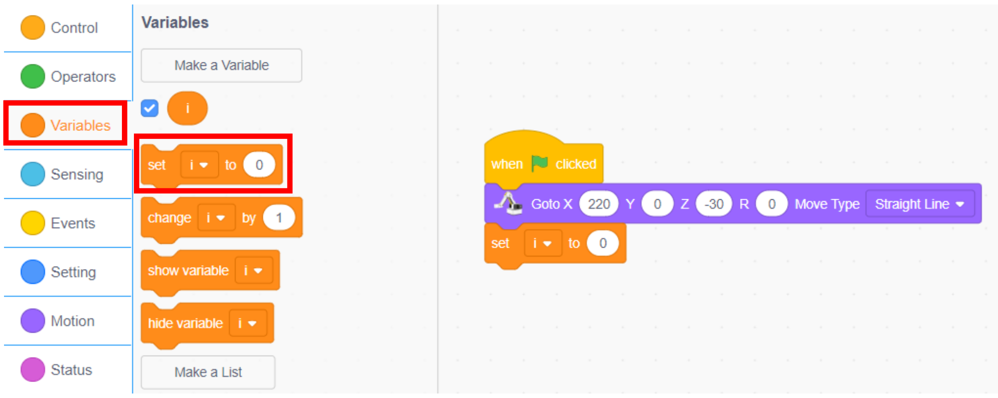


Figure 2.6 Setting the variable

(1) When you set the initial value of a variable, can you set it to any value?

(2) If the initial value is changed, will it affect the subsequent value setting?

Step 7: Enable the robotic arm to move upward.

1) Click the "Motion" tab and drag the "Relative Move" block to the code area, and put it together it with the previous block, as shown in Figure 2.7.

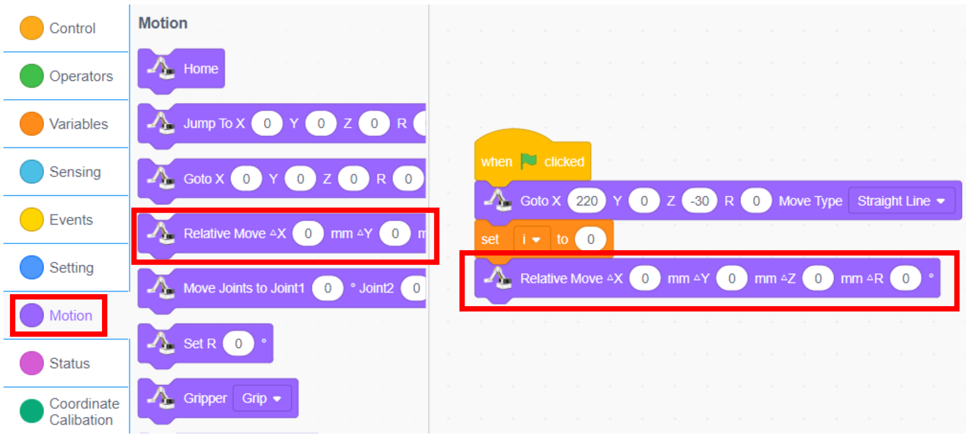


Figure 2.7 Relative motion blocks

2) Click the "**Variables**" tab, drag the variable "i" to the code area, and put it together at the oval position after △Z, as shown in Figure 2.8.

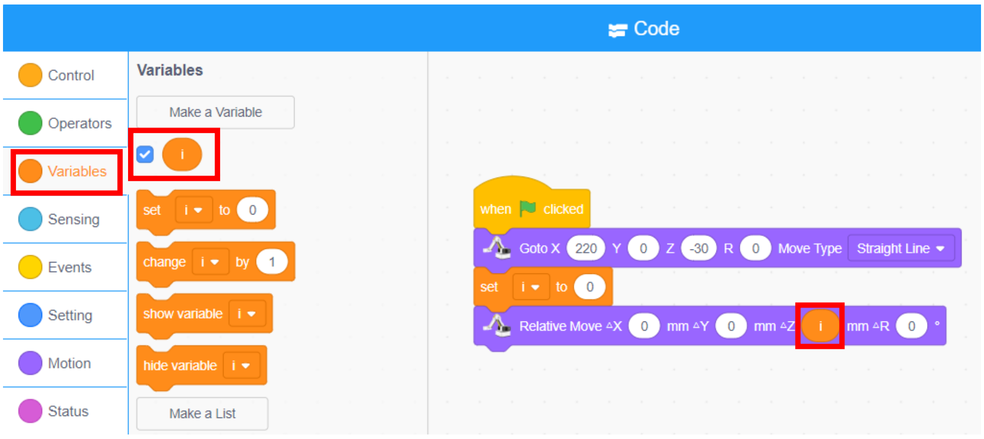


Figure 2.8 Variable "i" block

Step 8: Increase the variable i. Each time, the robotic arm moves upward by the same distance. If the robotic arm moves upward by 30mm, the variable i should be increased by 30. Click the "Variables" tab, drag the "Increase (i) by 1" block to the code area, and change the increment of the variable to 30, as shown in Figure 2.9.

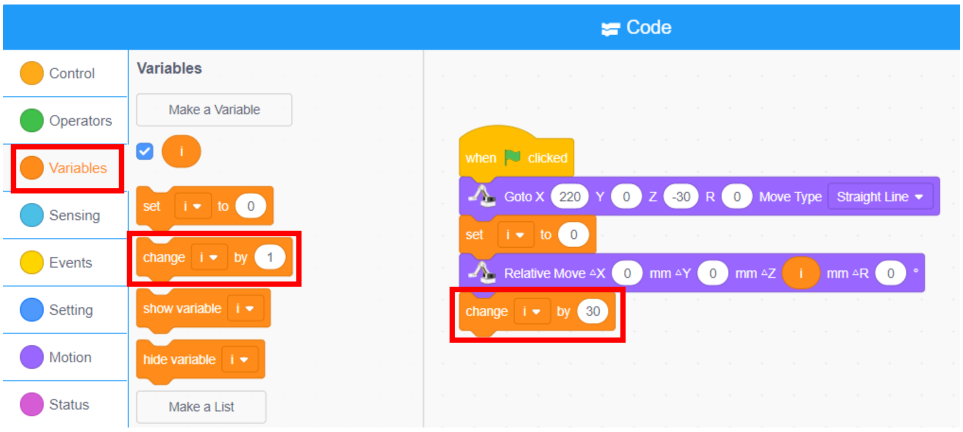


Figure 2.9 Block of increasing the variable

Step 9: Repeat the execution. In order to make the robotic arm move upwards 3 times repeatedly by a same distance, here we will use the repeated execution block.

Click the "**Control**" tab, drag the "Repeat (10)" block to the code area, put it under the "Set (i) to 0" block, and then change the number of repeated executions to 3, as shown in Figure 2.10.

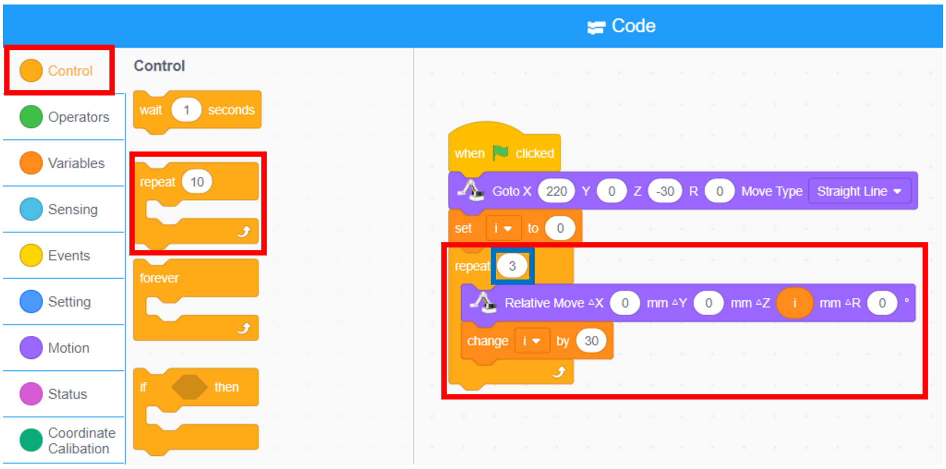


Figure 2.10 Block of repeated execution

Step 8: After completing the upward movement by the robotic arm three times, return the robotic arm to the home position.

In Scratch, try to find the homing block and put it together with the previous block.

Step 9: Integrate the process. Integrate the program of repeating the upward movement 3 times by the robotic arm, as shown in Figure 2.11.

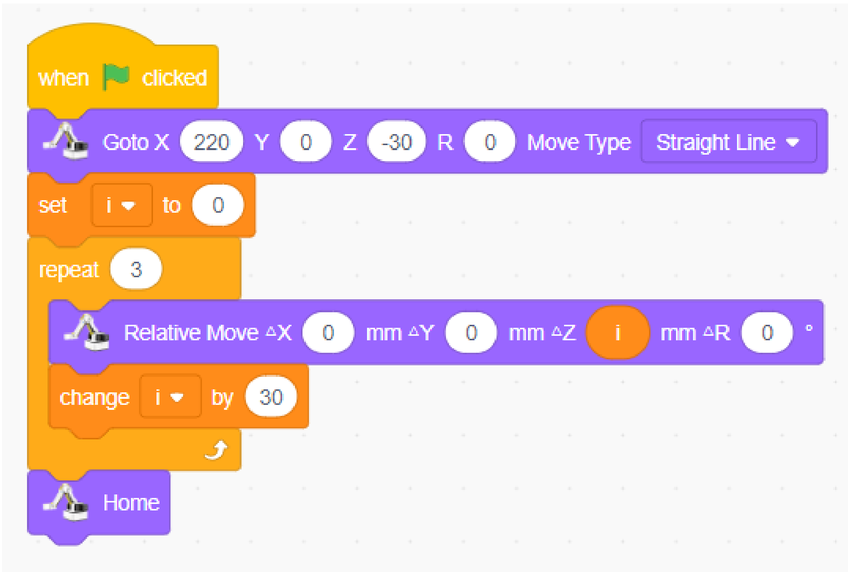


Figure 2.11 Process integration

3. Summary

(1) When creating a variable, it is used for____roles. (Fill in: all, current)

(2) When the robotic arm moves forward repeatedly, the block of variable "i" should be placed in the oval position after the___in the block of relative movement (Fill in: △X, △Y, △Z)

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I’ve completed the operation of creating a variable	
I’ve completed the operation of setting the variable	
I’ve known the principle of repeated execution	
I’ve completed the programming task of repeating upward movement 3 times by the robotic arm	

Task 2: Apply the Formula

In Scratch, there are many types of blocks related to repeated execution and now you are already proficient in one of them. Now, you will learn another repeated execution block in Scratch and apply the formula to make the robotic arm repeat the forward motion 4 times.

1. Observations

First, observe how the teacher makes the robotic arm repeat the forward movement 4 times. Then, fill in the blank in the table below according to your observation.

During the process of repeating forward motion 4 times by the robotic arm, the coordinates_______(fill in: X, Y, Z) of the robot arm change, and the coordinates status:________ (Fill in: increase, decrease, do not change).

2. Steps

(1) Prepare Hardware

Step 1: Connect Magician Lite to a power supply.

Step 2: Use a USB cable to connect Magician Lite to the computer and turn on Magician Lite.

(2) Design Program

Step 1: According to your observation, read the flowchart showing the robot arm move forward 4 times, as shown in Figure 2.12.

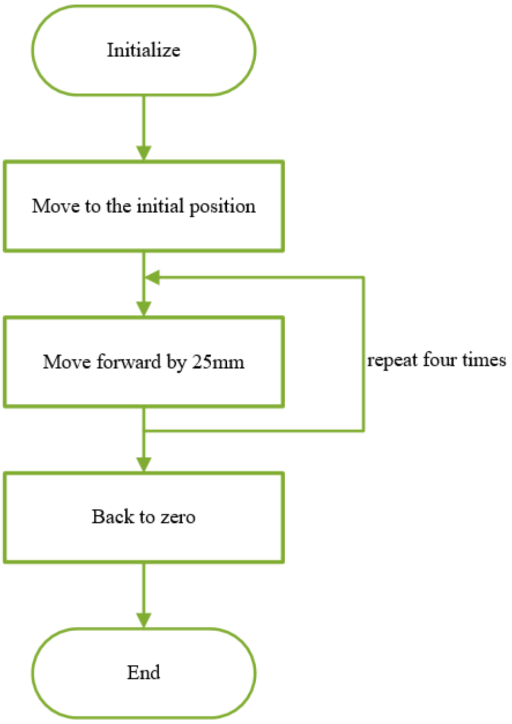


Figure 2.12 Flowchart of repeating forward motion

Step 2: Try to add the device Magician Lite in the software interface and connect to it.

Step 3: Find the starting block and drag it to the code area.

Step 4: Move to the initial position.

> ☛ In Scratch, try to move the robotic arm to the position (220, 0, 30), and select linear motion to move the robotic arm.

Step 5: Create a variable.

Review the steps in Task 1 to create a variable.

In Scratch, first, create the variable i (this variable is used for all roles), and then set the initial value of the variable i to 0.

Step 6: Enable the robot arm to move forward.

1) Click the "**Motion**" tab and drag the "Relative Move" block to the code area, and put it together it with the previous block, as shown in Figure 2.13.

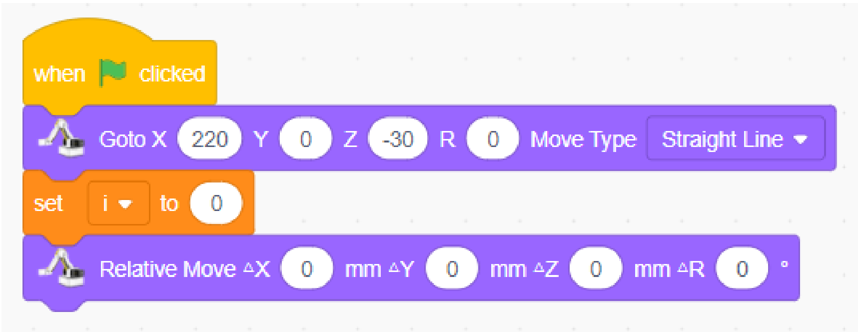


Figure 2.13 Relative motion blocks

2) Click the "**Operators**" tab, drag the operator "*" block to the code area, and put it in the oval position after △Z. Fill 25 on the left side of the operator "*" block, as shown in Figure 2.14

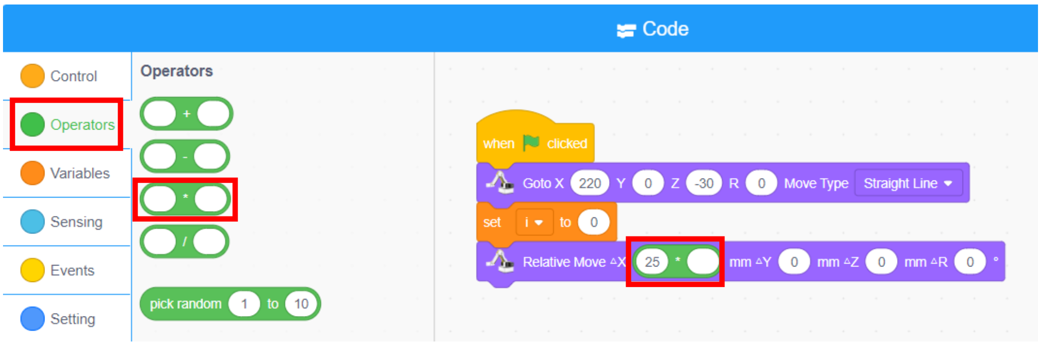


Figure 2.14 Block of operator "*"

We know that there are four types of mathematical calculation: addition, subtraction, multiplication, and division. In Scratch, there are four blocks for them. The operator "+" block represents the plus sign, the operator "-" block the minus sign, the operator "*" block the multiplication sign, and the operator " / " block the division sign.

3) Click the "**Variables**" tab, drag the variable "i" to the code area, and put it in the oval position to the right of the block of the operator "*", as shown in Figure 2.15.

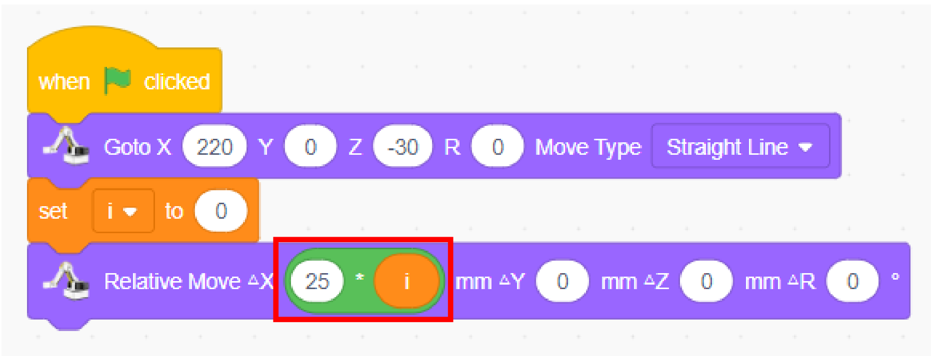


Figure 2.15 Variable "i" block

Step 7: Count the number of movements. Each time the robotic arm moves forward, the value of the variable i increases by 1. Click the "**Variables**" tab, drag and drop the "increase the variable" block to the code area and put the blocks together, as shown in Figure 2.16.

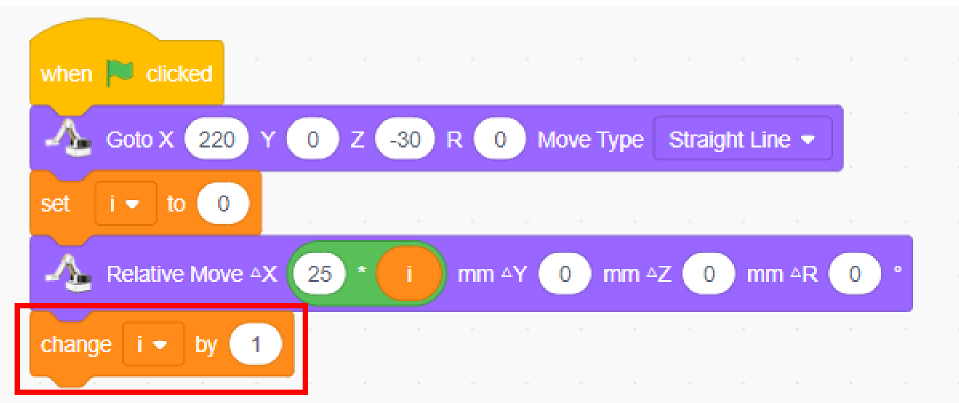


Figure 2.16 Block of increasing the variable

Step 8: Repeat execution.

1) You can use "repeat until" block. Click the "**Control**" tab and drag the "repeat until" block to the code area, as shown in Figure 2.17.

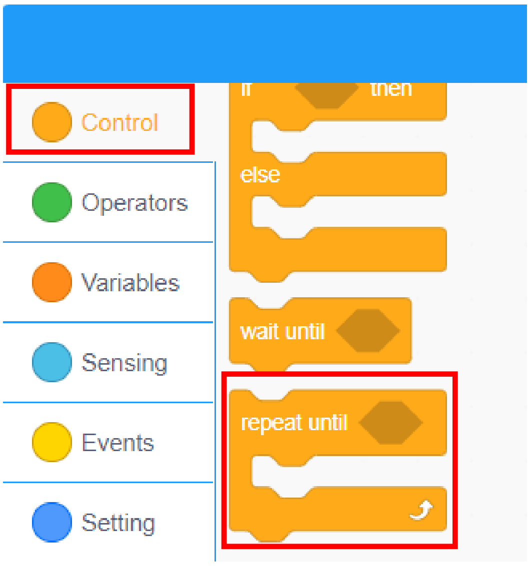


Figure 2.17 Block of repeated execution

The use of the block "repeat until" is shown in the figure:

2) Set the critical condition.

The robot arm moves forward 4 times repeatedly. In this case, is it 4 times that the critical condition should be set to repeat?

The variable i starts from 0, so the first forward movement is performed when i = 0, the second time is executed when i = 1, the third time is executed when i = 2, and the fourth time is executed when i = 3. Therefore, i = 4 is the critical condition for repeated execution.

In the Scratch interface, click the "**Operators**" tab, drag the operator "=" block to the code area, and combine it to the critical condition position of the block "repeat until".

Click the "**Variables**" tab and drag the "Variable i" block to the left of the "Equal Sign" block in the code area, and fill the number "4" to the right of the "Equal Sign" block, as shown in Figure 2.18.

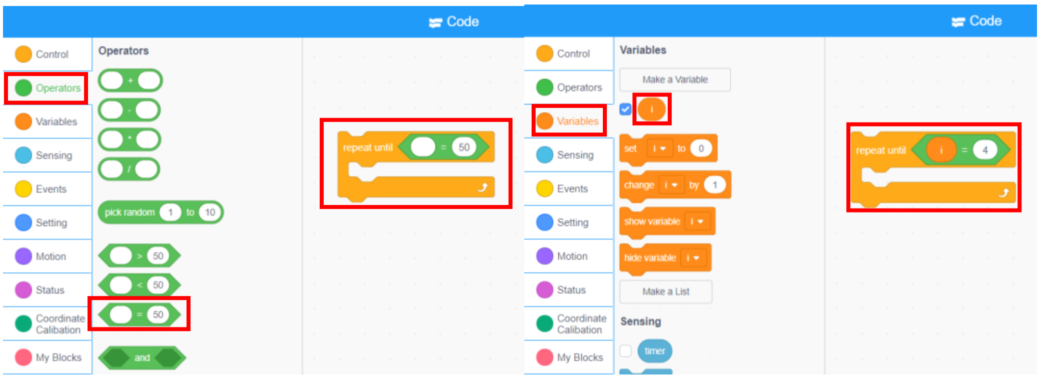


Figure 2.18 Block related to the critical condition

Step 9: Integrate the process. Put the "repeat until" block under the "Set (i) to 0" block, drag the "Home" block, and put it after the "repeat until" block. Integrate the program of the robot arm moving forward 4 times is shown in Figure 2.19.

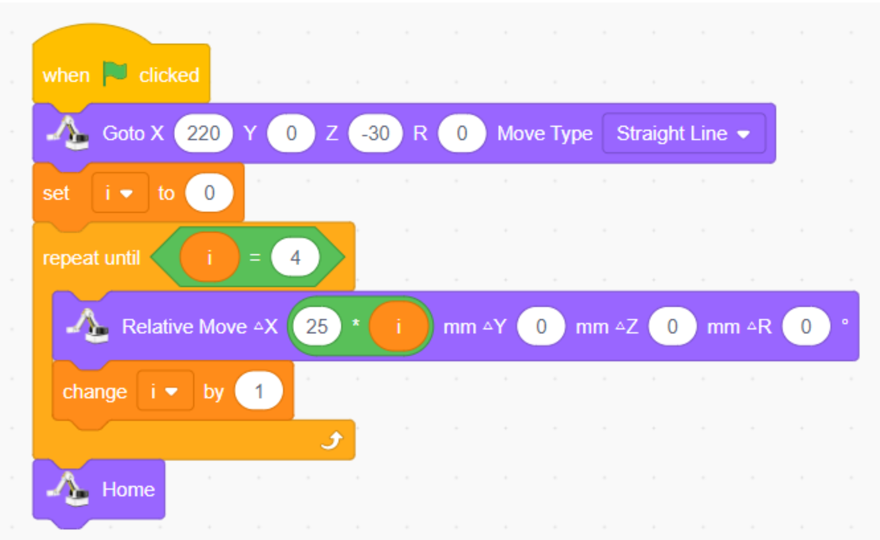


Figure 2.19 Process integration

3. Summary

(1) The main differences between "forever" blocks and "repeat until" blocks are: ____

____

(2) When the robot arm moves to the right repeatedly, the block of operator "*" should be placed in the oval position behind_____of the relative motion block. (Fill in: △X, △Y, △Z)

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I've created and set the variables	
I've known the principle and function of the formula	
I've known the principle of repeat execution until	
I've completed the programming task of repeating the forward movement of the robotic arm 4 times	

Task 3: Transport Two Objects

When a porter transports an object, the porter usually goes to one location, picks up the object, then goes to another position and drops it. You will program in Scratch, and enable the robotic arm to complete the process as well as to transport two objects.

1. Observations

First, observe how the teacher makes the robotic arm transport two objects. Then, fill in the blank in the table below according to your observation.

There are ____________ fixed positions for the robot arm to transport objects.

The height of the object at the initial position: the height is ___________ as the number of transportation increases (fill in: higher, lower, unchanged); the height of the object at the target position: the height is ___________ than/as the number of transportation increases (fill in: higher, lower, unchanged);

2. Steps

(1) Prepare Hardware

Step 1: Design the experiment map and place the equipment. Refer to Figure 2.20 for the experiment map.

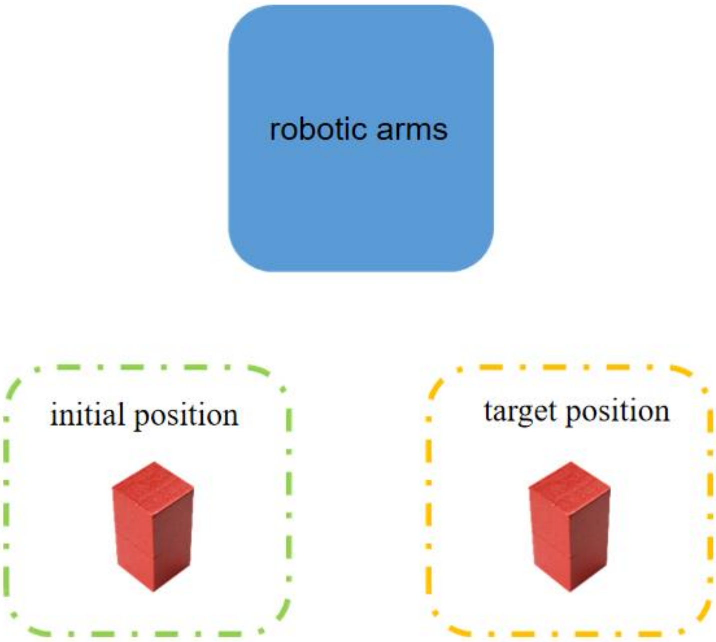


Figure 2.20 Experiment map

Step 2: Connect Magician Lite to a power supply.

Step 3: Use a USB cable to connect Magician Lite to the computer and turn on Magician Lite.

(2) Design Program

Step 1: According to your observation, read the flowchart of moving upwards 4 times by the robot arm, as shown in Figure 2.21.

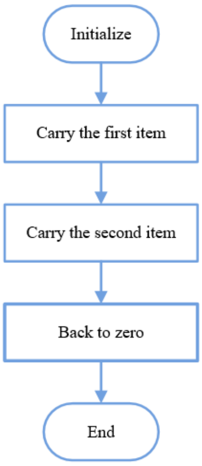
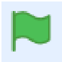


Figure 2.21 Flowchart of transporting two objects

Step 2: Try to add the device Magician Lite in the software interface and connect it.

Step 3: Find the starting block and drag it to the code area.

Step 4: Transport the first object.

1) Move to the first object at the initial position. Here the position of the first object is recorded as (260, -50, -17). Click the "**Motion**" tab, drag the “Jump To" block to the code area, and put it under the "when  is clicked" block, as shown in Figure 2.22.

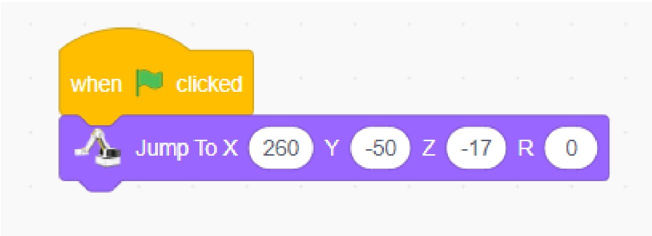


Figure 2.22 Jump to the position of the first object

The robotic arm moves to the position of the first object (260, -50, -17), it can be determined according to the specific placement. At the same time, it can be found that the height of each object is 25mm. When the two objects are neatly stacked, the Z coordinate value of the top surface of the top object is -17.

2) After moving to the position of the first object, delay 0.5 seconds. Click the "**Control**" tab and drag the "wait (1) seconds" block to the code area, change the waiting time to 0.5 seconds, and put it together with the previous block, as shown in Figure 2.23.

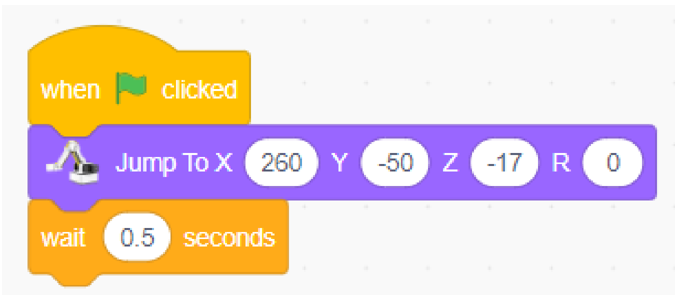


Figure 2.23 The blocks of waiting

3) Grab objects. Click the "**Motion**" tab, drag the "Suction Cup" block to the code area, and put it under the "wait (0.5) seconds" block, as shown in Figure 2.24.

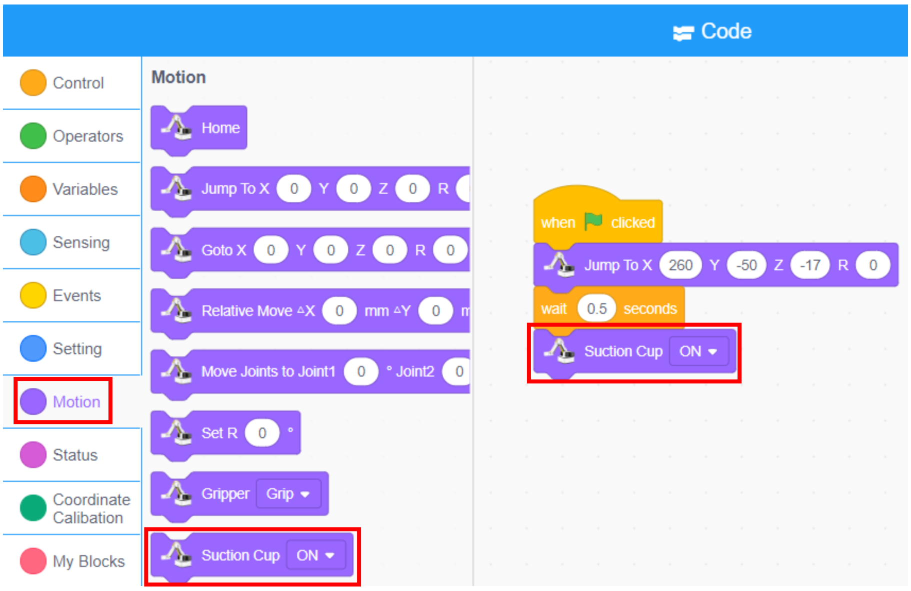


Figure 2.24 Grabbing objects

4) Move to the target position and drop the object.

Which blocks are included from moving to the target position to dropping the objects?

In Scratch, try to Jump move the arm to (260, 50, -43), wait 0.5 seconds, and then place the object (that is, the suction cup is released).

Step 5: Move the second object.

Review which steps are involved in grabbing an object with a robotic arm

In Scratch, try to program a robotic arm to carry the second object.

(1) JUMP move to the second object (260, -50, -43) in the initial position, waits 0.5 seconds, and then grabs the object (that is, the suction cup is in grasping state).

(2) JUMP move to the second object (260, 50, -17) at the target position, waits 0.5 seconds, and then places the object (that is,

3. Summary

(1) The key steps involved in the robotic arm's handling of two objects are: ____

____

(2) The same blocks involved in the program of carrying two objects are: ____

____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion
I’ve known the steps to grab and place objects	
I’ve used suction cup control instructions	
I’ve completed the programming task of grabbing and placing objects	
I’ve completed the programming task of the robotic arm transporting two objects	

Task 4: Transport Four Objects

Through Task 3, you have made the robotic arm transport two objects already. It is not difficult to find that in Scratch programming, the block used to transport the first object is the same as that used to transport the second object. Now, you will use the repeated execution, create a variable, and apply the formula in Scratch programming to enable the robotic arm to transport four objects.

1. Observations

First, observe how the teacher makes the robotic arm carry four objects. Then, fill in the blank in the table below according to your observation.

In Scratch programming, the repeat executing blocks used are: (Fill "repeat (10)" block, "repeat until" blocks).
The pattern of how the height of the object changes at the initial position:
The pattern of how the height of the object changes at the target position:

2. Steps

(1) Prepare Hardware

Step 1: Design the experiment map and place the equipment. Refer to Figure 2.25 for the experiment map.

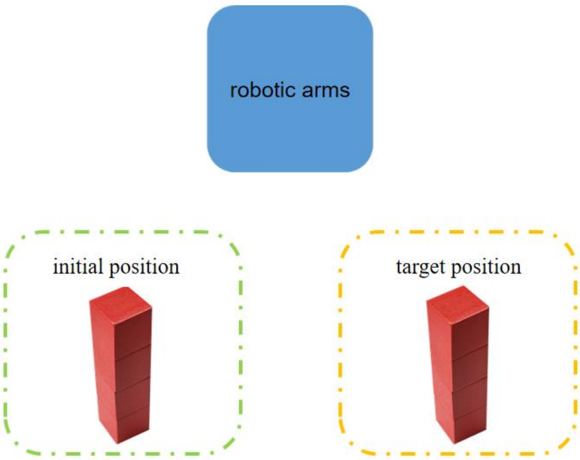


Figure 2.25 Experiment map

Step 2: Connect Magician Lite to a power supplier.

Step 3: Use a USB cable to connect Magician Lite to the computer and turn on Magician Lite.

(2) Design Program

Step 1: According to your observation, read the flowchart of the robot arm transporting 4 times, as shown in Figure 2.26.

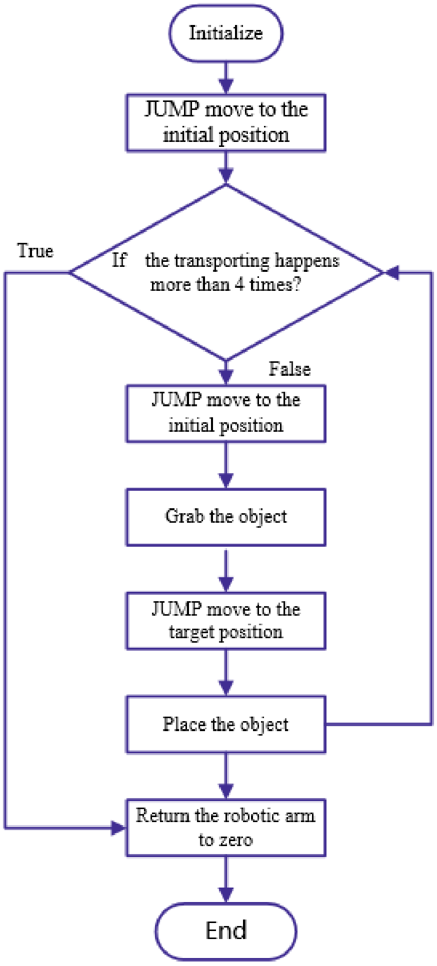


Figure 2.26 Flowchart of transporting 4 objects

Step 2: Add and connect to the device in software interface.

In Scratch, try to add the device Magician Lite and connect to the device.

Step 3: Start and create a variable.

In Scratch, find the block of starting and drag it to the code area. Create a variable i and set the initial value to 0.

Step 4: Transport one object.

In Scratch, observe and test the height of the object at the initial position and the target position through the control panel.

When you neatly stack 4 objects, the position of the upper surface of the uppermost object at the initial position is (260, -50, 57), and the position of the upper surface of the lowermost object at the target position is (260, 50, -43). Note that during operation, you can decide according to the specific placement situation.

When the robotic arm transports the first object, JUMP move to the upper surface of the first object in the initial position to grab the object; then Jump move to the target position to place the object. The programming method is shown in Figure 2.27.

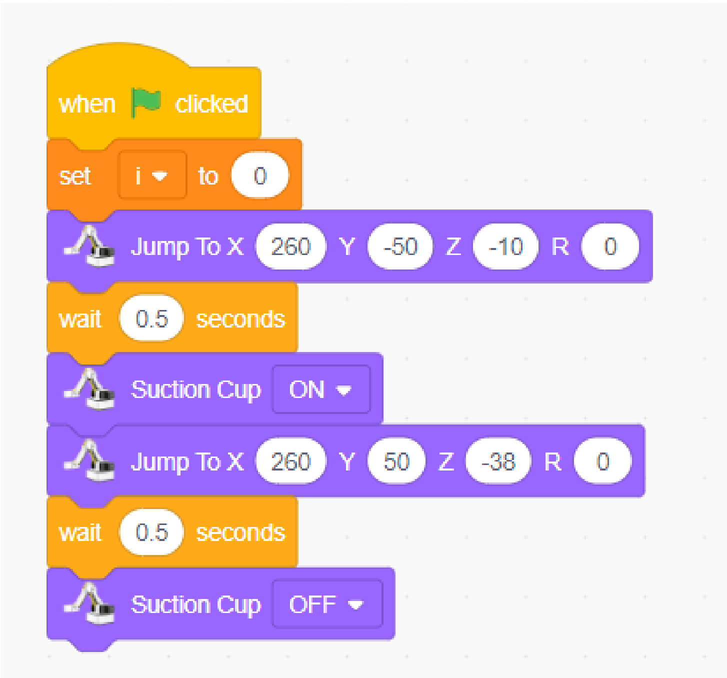


Figure 2.27 Transporting one object

Step 5: Apply the formula.

1) Click the "**Operators**" tab, drag the "-" operator block to the code area, put it in the Z-coordinate position of the first "Jump To" block, and fill in the left side of the operator "-" block with 57 as shown in Figure 2.28.

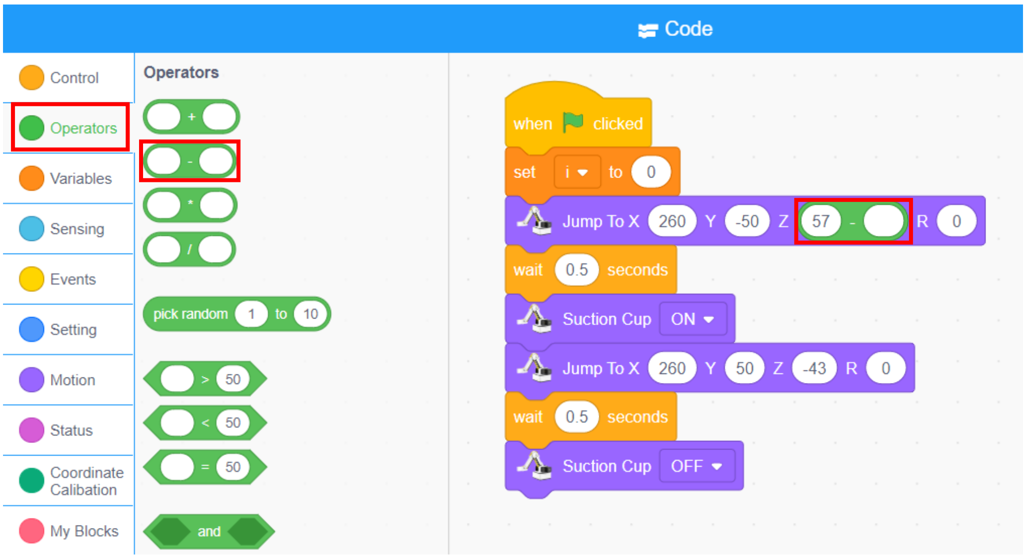


Figure 2.28 Operators"-" blocks

2) Click the "Operators" tab, drag the operators "*" block to the code area, splice it at the right elliptical position of the operators "-" block, and fill in 25 to the left of the operator "*" block, as shown in the figure As shown in 2.29.

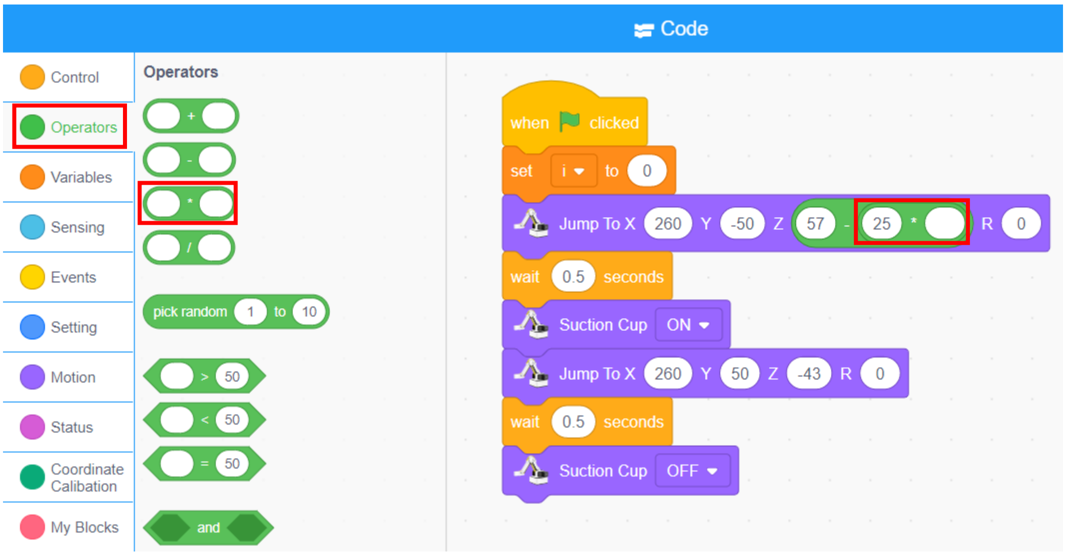


Figure 2.29 Operators "*" blocks

3) Click the "**Variables**" tab, drag the variable "i" to the code area, and fill it in the right oval of the operator "*", as shown in Figure 2.30.

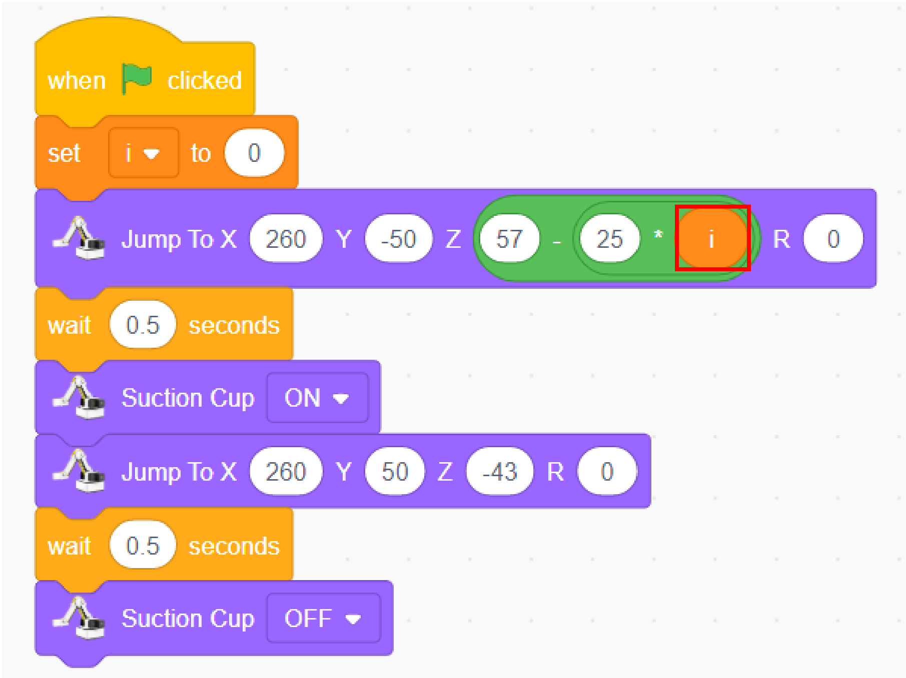


Figure 2.30 Variable "i" blocks

4) Formula for height change of target position.

While the robotic arm is carrying four objects, talk about the height change of the target position. The height change formula of the target position is calculated based on the height change formula of the initial position.

In Scratch, the formula for changing the target position is applied to the Z-coordinate position of the second Jump move block programmatically.

Step 6: Increase the variable i. Click on the "**Variables**" tab and drag the "change （i） by 1" brick to the code area, and put it together with the previous block, as shown in Figure 2.31.

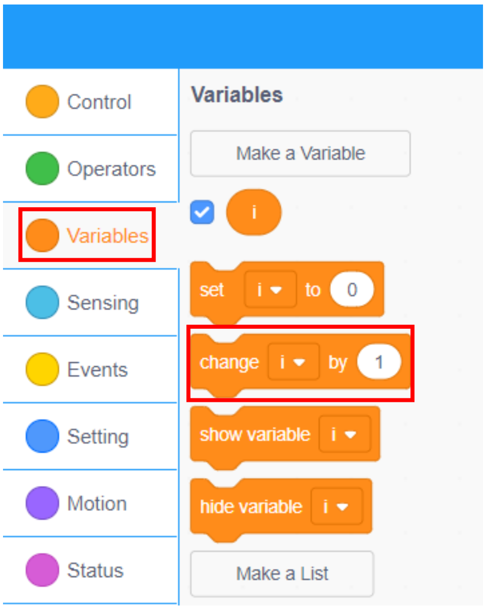


Figure 2.31 Variable i increased

Step 7: Repeat transporting four objects. This experiment task uses the "repeat until" blocks. Since four objects are being carried, the judgment condition is: i = 4.

Put the "repeat until" block under the "set (i) to 0" block, drag the "Home" block, and put it after the "repeat until" block. Integrate the program of transporting 4 objects by the robotic arm, as shown in Figure 2.32.

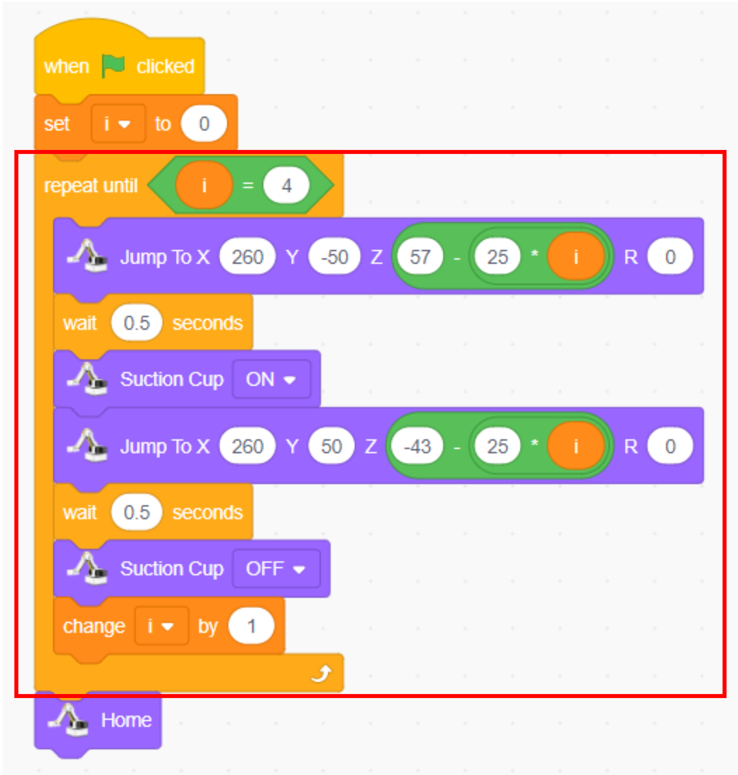


Figure 2.32 Program integration

3. Summary

(1) While the robotic arm is carrying four objects , the height changes of the initial position and the target position are: ______________

____

(2) Write down the height change formula of the initial position and the target position: ______________________________

____

(3) Role of using repeat executing blocks: ______________________

____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I've created a variable and used it in a task	
I've known the pattern of the height changing of the object at the initial and target position:	
I've applied the instruction of the height changing formula	
I've used repeated execution instructions	
I've completed the programming task of transporting four objects by the robotic arm	

Experiment 3

Garbage Classification

Team Name: Team member: Date:

Overview

Paper, plastic, rind, old batteries, etc. are common trashes in campuses. With the continuous improvement of people's living standards, the composition of campus garbage is becoming increasingly complex. To dispose of such campus garbage, first you need to classify them. Some people use artificial intelligence (AI) to classify garbage automatically. This experiment simulates the process of automatic garbage classification, and uses image recognition technology to realize automatic classification of garbage by a robotic arm. Now, let's complete this experiment.

Objectives

1. Understand the image data collection and training process by performing garbage data collection and training.
2. Master the image identification instruction by completing "Image Identification".
3. Master the application of voice playback by completing the "Add Voice Prompt Function".

Equipment

Icon	Name	Quantity
	Dobot Magician Lite robotic arm	1

Icon	Name	Quantity
	Suction cup kit	1
	Type-C cable	1
	Power adapter	1
	Garbage image paperboards	Several
	Garbage bin	4

Experiment Requirements

- Take care when using electricity.
- Before the experiment, check whether the experiment equipment is complete and intact. If there is any omission or damage, please report to the teacher.
- Any specific operation in the experiment should be carried out according to the experiment manual. If you have any questions, please promptly ask the teacher.
- In the experiment, the joint of the robotic arm is in a working state after being powered on. Do not force it to move without pressing the unlock button.

- Equipment failure during the experiment must be reported to the instructor in time. Do not handle it by yourself.
- After the experiment, the equipment should be arranged well. You are allowed to leave the laboratory only after the inspection by the group leader.

Task 1: Collect Garbage Data and Train Collect

Garbage data collection and training, adding labels for four types of garbage and collecting pictures corresponding to the four types of garbage, training various types of garbage models, and testing garbage classification models.

1. Observations

Please observe the teacher's steps to collect and train garbage data, and complete the remaining steps based on observations and tips.

> Create new classification data: add features and data ⟶ ____________
>
> ____
>
> ____
>
> ____
>
> (Follow the prompts to complete the remaining steps)

2. Steps

Step 1: Open the DobotScratch software, select the device Magician Lite and connect it, and add the AI Extension module, as shown in Figure 3.1.

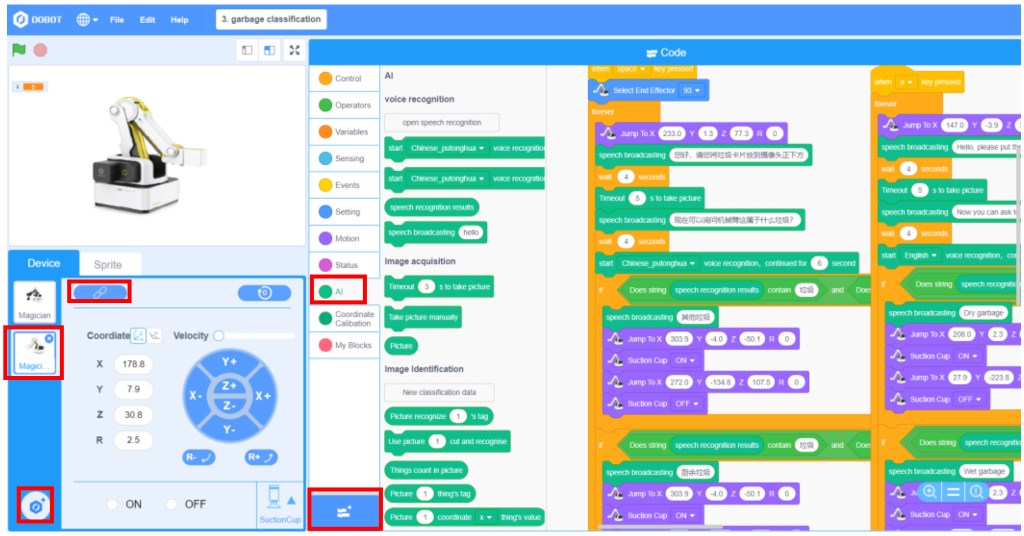


Figure 3.1 Software settings

Step 2: Create new classification data, as shown in Figure 3.2.

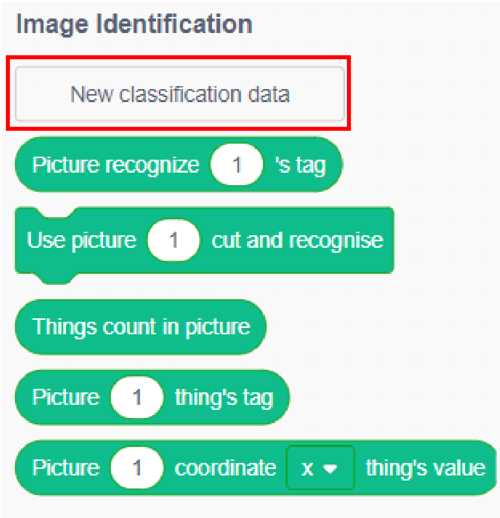


Figure 3.2 Create new classification data

Step 3: Add features and data (that is, collect garbage images), take pictures of the types of garbage that need to be identified, and store them, as shown in Figure 3.3.

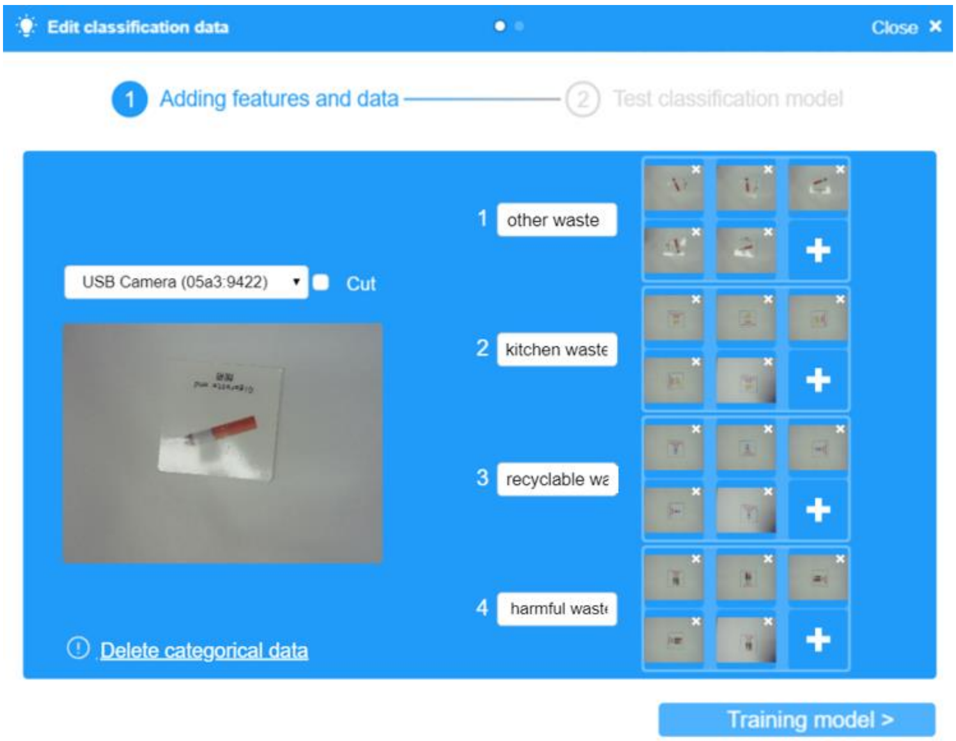


Figure 3.3 Garbage image collection

Step 4: Train the model. The process of garbage image preprocessing and feature extraction is the process of computer learning. Figure 3.4 shows the model training.

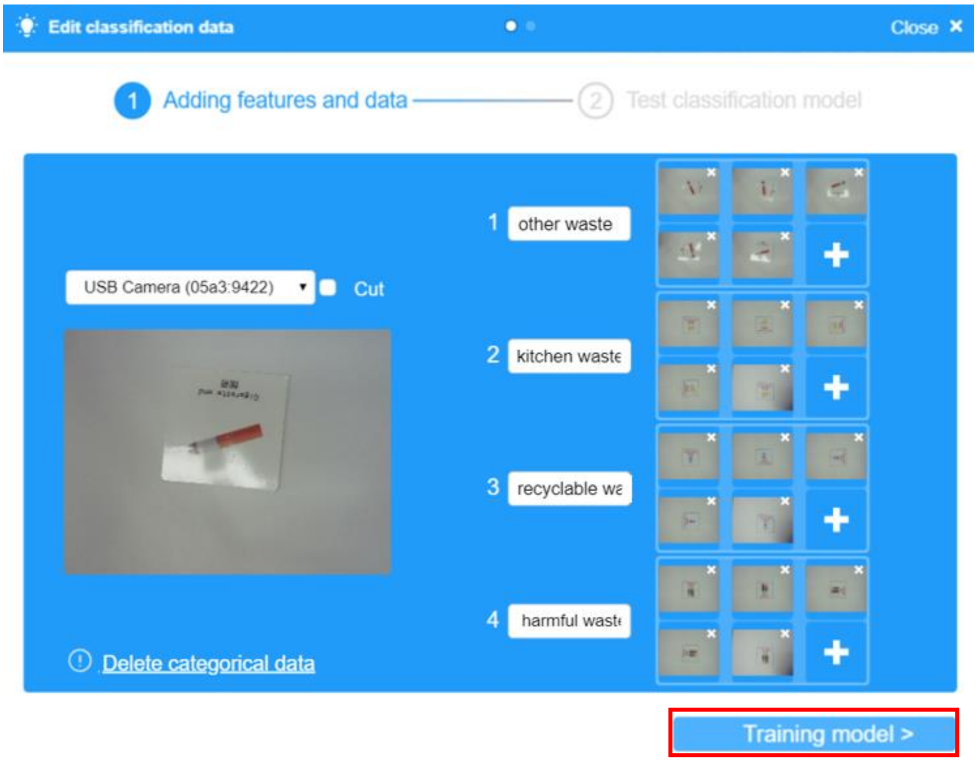


Figure 3.4 Training the model

Step 5: Test classification model, place all types of garbage under the camera, and test the accuracy of recognition, as shown in Figure 3.5.

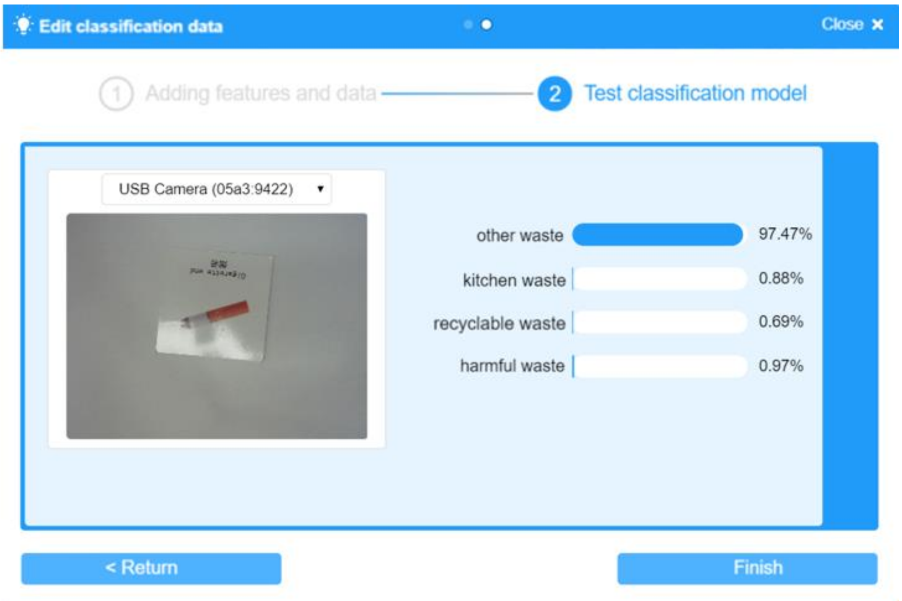


Figure 3.5 Test classification model

Step 6: Click "**Finish**" to save the file, as shown in Figure 3.6. Then, you can start writing the program.

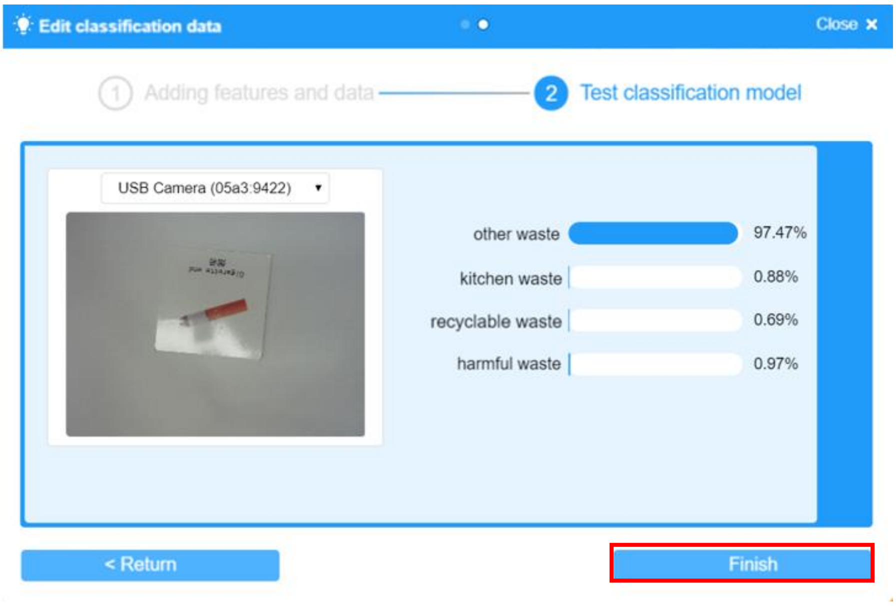


Figure 3.6 Clicking "Finish"

☛ Implement the following through programing: If the image recognition result is "other waste", the voice "other waste" is played.

3. Summary

Summarize the steps of garbage data collection and training:________

____

____

____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) the unfinished ones.

Assessment Content	Completion Status
I've completed the task of adding garbage features and data	
I've completed the task of testing the classification model	

Task 2: Identify Image

In task 1, we have collected and trained garbage images, and divided the garbage image into four types. In task 2, we can use the trained garbage model directly to classify garbage.

1. Observations

Please observe how the garbage classification system works and follow the prompts to complete the general steps of Design Program.

Design the program: Set the end tool to suction cup > initialize the end position of the robotic arm > collect garbage images > identify the collected garbage images

(Follow the prompts to complete the remaining steps)

2. Steps

(1) Prepare Hardware

Step 1: After preparing the equipment, power on the robotic arm and connect the robotic arm to the personal computer (PC).

Step 2: Place the equipment on the corresponding position on the map, as shown in Figure 3.7, adjust the robotic arm, and turn it on.

Figure 3.7 Placement of equipment

(2) Design Program

Step 1: Analyze the implementation process of garbage classification and draw a flowchart, as shown in Figure 3.8.

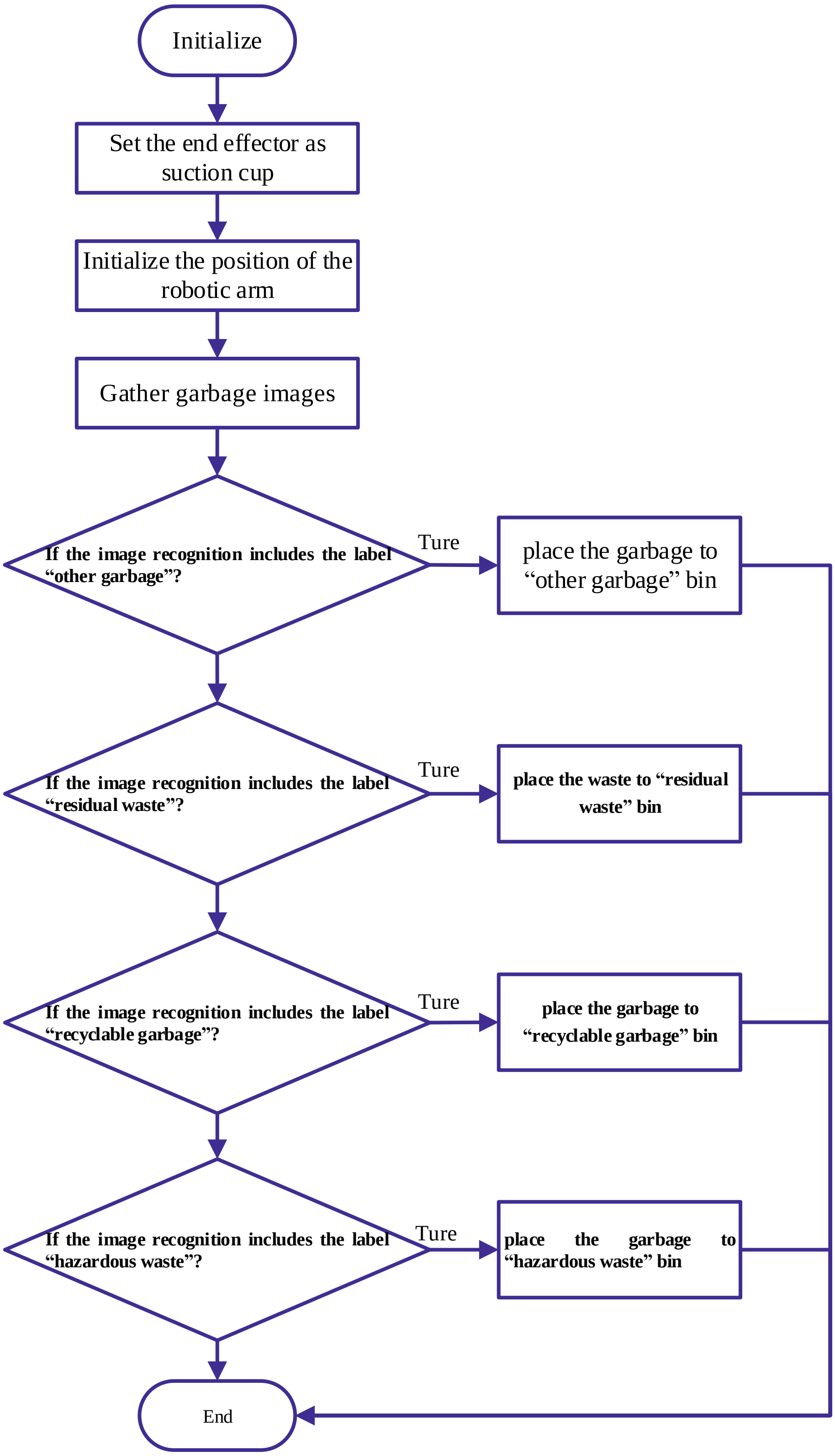


Figure 3.8 Flowchart

Step 2: Open the program file saved in the previous lesson, and start programming.

Step 3: Select "When space key pressed" to trigger the program to run, select "Select End Effector Suction Cup" to set the initial position of the end of the robotic arm, as shown in Figure 3.9.

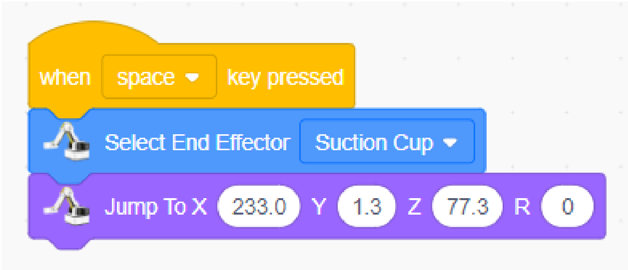


Figure 3.9 Initialization

Step 4: Select the "Timeout (3)s to take picture" block to gather garbage images, as shown in Figure 3.10

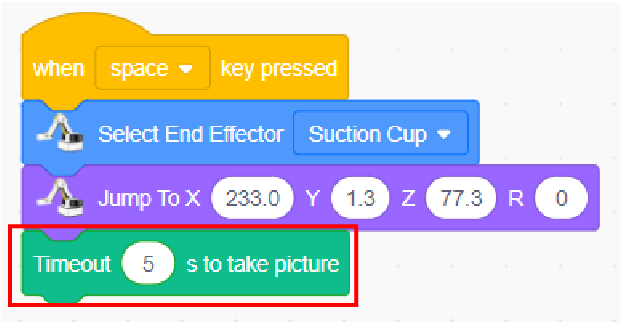


Figure 3.10 Garbage image collection

Step 5: Set the conditions for the robotic arm to classify the garbage, that is, to determine the image recognition result, as shown in Figure 3.11.

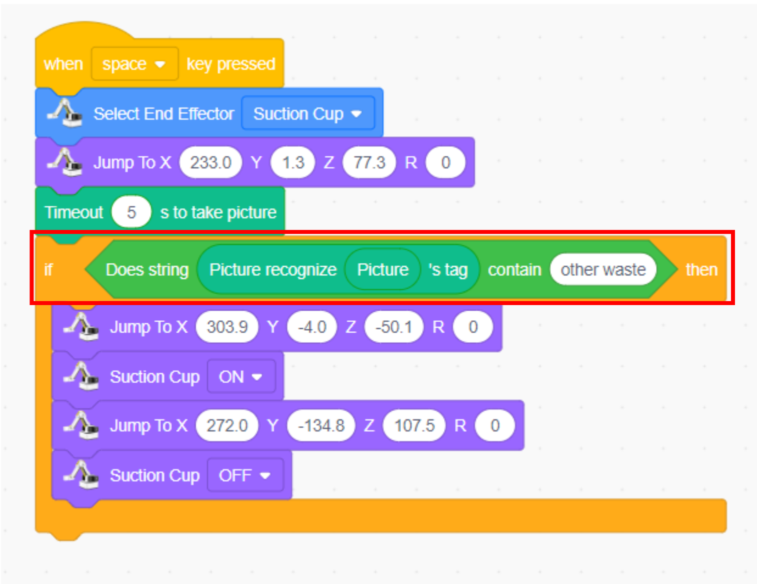


Figure 3.11 Setting the conditions for the robot to classify garbage

Step 7: If the conditions are met, place the garbage in the corresponding garbage bin, as shown in Figure 3.12.

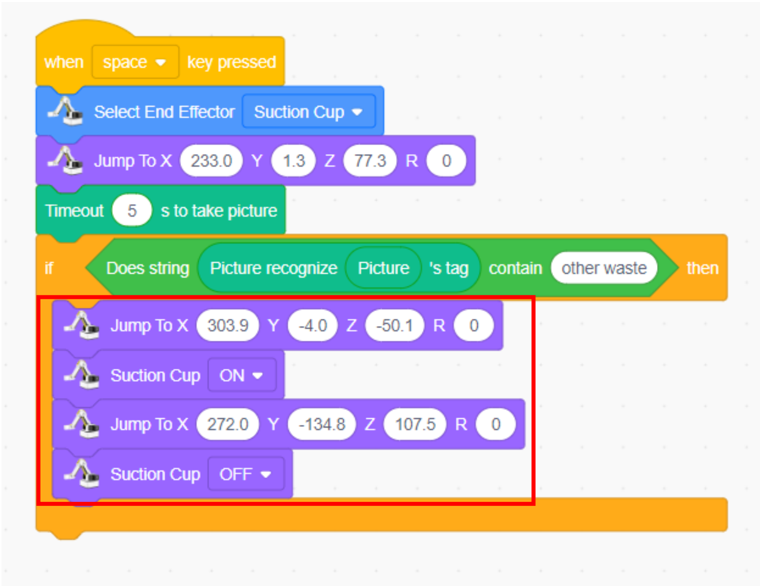


Figure 3.12 Robotic arm automatically classifies garbage

In addition to using the "if ... then ..." block to achieve condition judgment, what other methods can we use?

Step 8: There are four types of garbage, so you need to judge it four times. The logic of the blocks is similar. The difference is the type of garbage and the position of the garbage bin. Here is an example of judging two types of garbage. The reference program is shown in Figure 3.13.

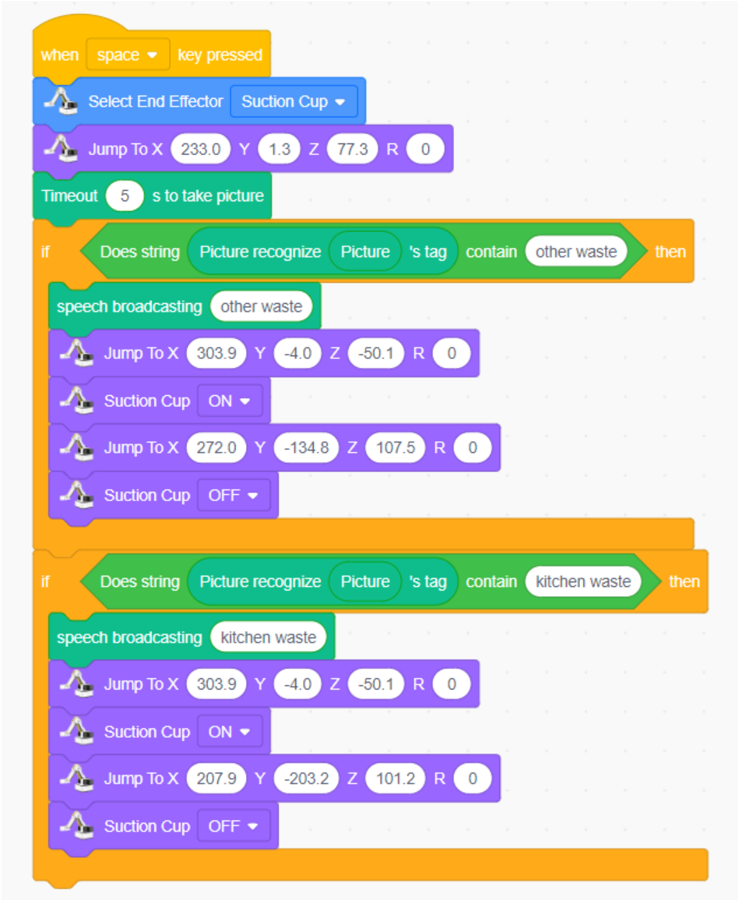


Figure 3.13 Garbage classification

☛ When running the program, you can find that the program executes the judgment only once. Please try to use the "forever" block to achieve multiple judgments.

3. Summary

____

____

____

____

(Summarize the knowledge you've learned in this experiment)

4. Self-Assessment

Check the content completed in the experiment task, tick (√) the finished ones, and circle (○) the unfinished ones.

Assessment Content	Completion Status
I've completed the task of initializing the end tool	
I've set the location where garbage images are collected	
I've completed the process of garbage classification once	
I've completed the process of garbage classification multiple times	
I've answered the questions raised in the experiment steps carefully	

Task 3: Voice Prompts

Add the voice prompt on the basis of task 2 to allow users to better use the garbage classification system.

1. Observations

Please observe how the system works after adding the voice prompt function, and complete the steps of Design Program based on your observation.

> Design the program: Set the end tool to suction cup > initialize the end position of the robotic arm > collect the garbage image > identify the collected garbage image > play "Other waste" if the image recognition label > put the garbage into the "Other waste" garbage bin. ________
>
> ____
>
> ____
>
> (Follow the prompts and complete the remaining steps)

2. Steps

Step 1: Prepare hardware. Refer to task 2 for details.

Step 2: Analyze the implementation steps of adding voice playback and voice recognition functions, and draw a flowchart, as shown in Figure 3.14.

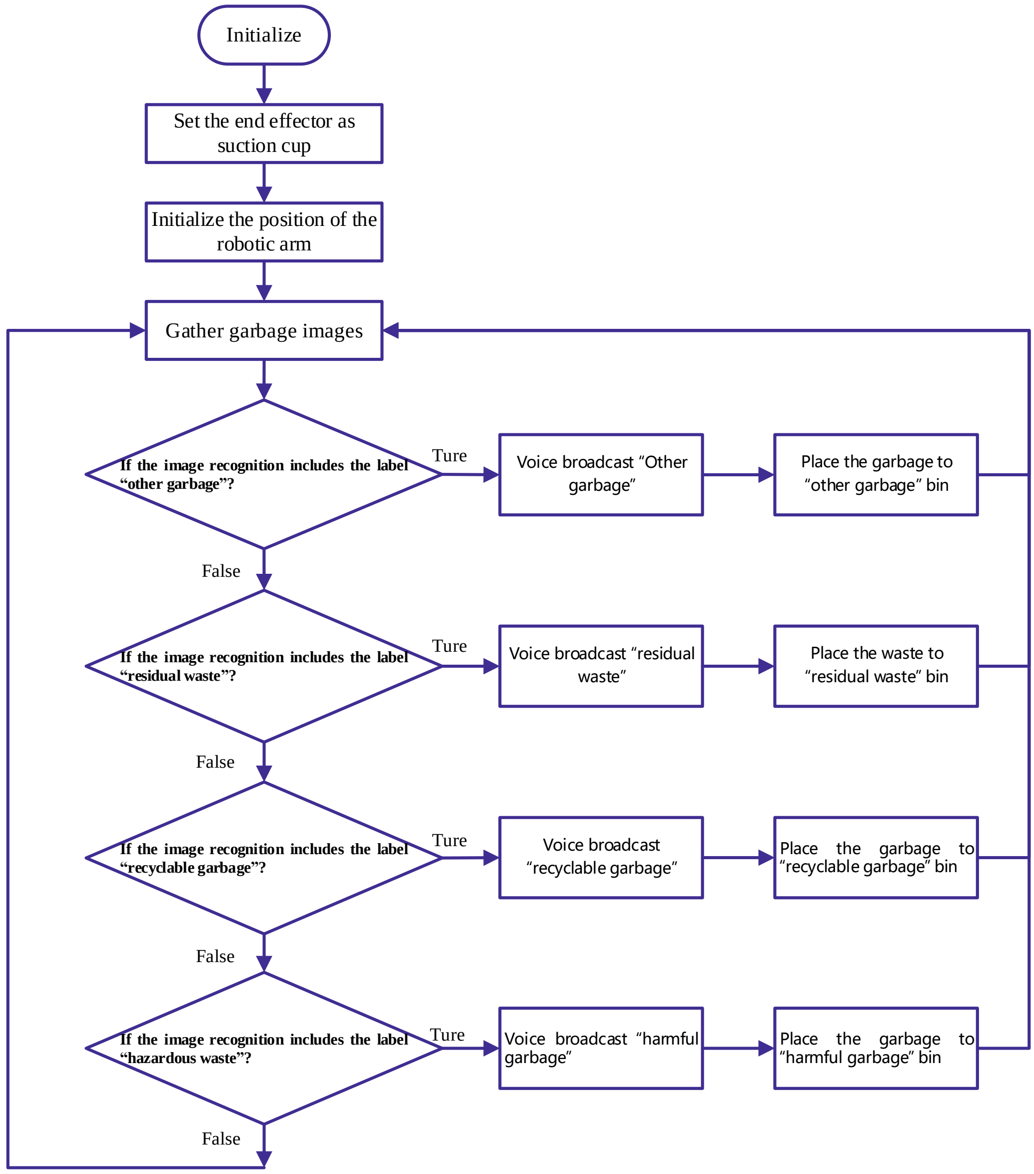


Figure 3.14 Flowchart

Step 3: Select the "speech broadcasting (hello)" block, and the voice prompt "Hello, please put the garbage card directly under the camera", where the voice playback takes about 4 seconds. Select the "wait (1) seconds" block, and change 1 second to 4 seconds, as shown in Figure 3.15.

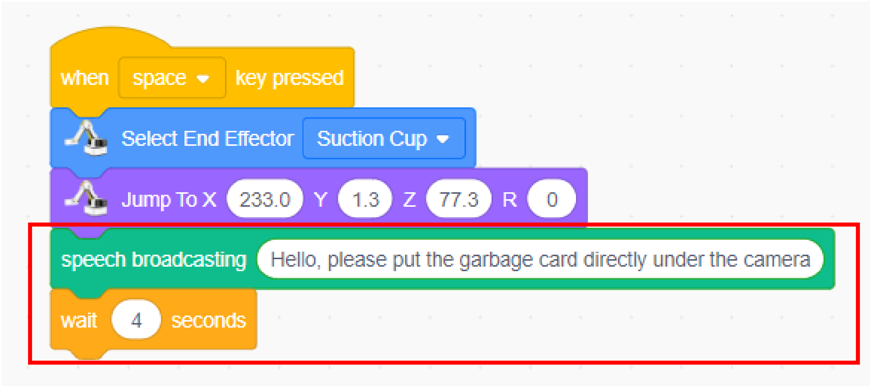


Figure 3.15 Voice prompt

Step 4: Select "speech broadcasting (hello)" to realize the voice prompt "You can now ask the robotic arm what type of garbage it is". About 4 seconds are reserved for the voice playback. Figure 3.16 shows the voice prompt.

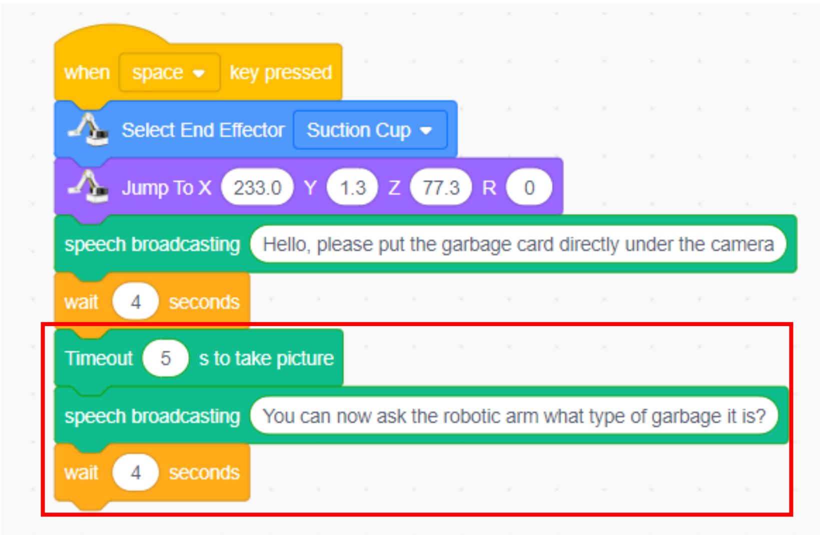


Figure 3.16 Voice prompt

Step 5: If you have all the conditions met, then play the voice prompt and put the garbage into the corresponding garbage bin, as shown in Figure 3.17.

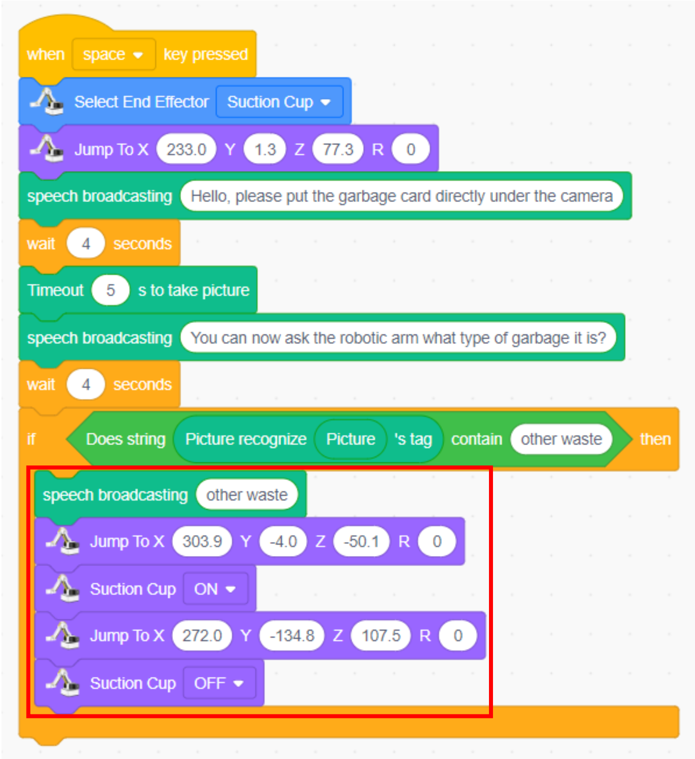


Figure 3.17 Robotic arm automatically classifies garbage

Step 6: There are four types of garbage. Here two types of garbage are used as an example for judgement. The program reference is shown in Figure 3.18.

Figure 3.18 Judging two times

Step 7: Include the “forever” block in the program because the garbage needs to be classified more than once. The program reference is shown in Figure 3.19.

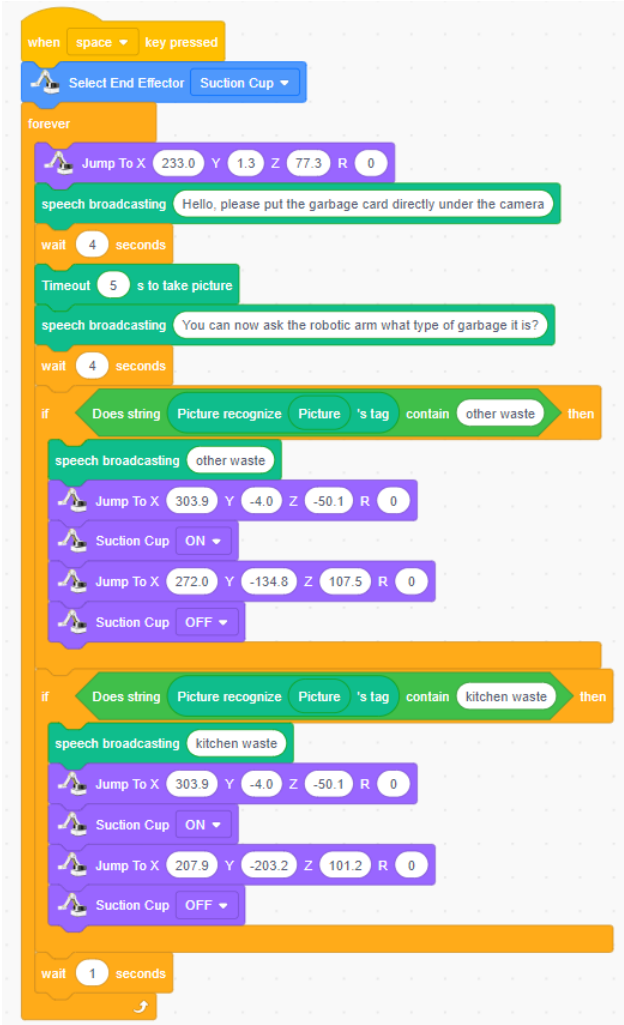


Figure 3.19 Garbage classification program

The program can only classify two types of garbage. Please modify the program to classify four types of garbage.

3. Summary

____

____

____

____

(Summarize the knowledge you've learned in this experiment)

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I've completed the task of adding "Voice prompt function"	
I've completed the questions raised in the experiment steps	

Experiment 4

Desk Arrangement

Team Name: Team member: Date:

Overview

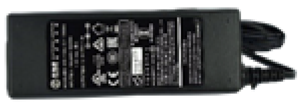
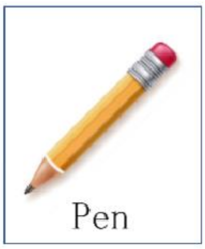
When students do their schoolwork, they often use pencils, rubber and other stationery. After completing schoolwork, they can use our robotic helper to help arrange their stationery and keep their desks neat and tidy. In this experiment, we will simulate a scenario, where a robot arranges sundries on the desk. This experiment aims to help students experience how the image recognition technology and the image segmentation technology are used. This experiment falls into four parts: the training model, the image recognition and segmentation, the desk arrangement, and the use of a multi-branch structure.

Objective

- Master how to train the image segmentation model in DobotScratch by training the image segmentation model.
- Understand how to use the instructions of image segmentation by programing the desk arrangement.
- Understand how to distinguish the global variables from local variables by programing the desk arrangement.
- Master what a multi-branch structure is by programming the desk arrangement,

Equipment

Illustrated Equipment	Name	Quantity

	Dobot Magician Lite Robotic arms	1
	USB Type-C interface line	1
	Suction cup	1
	Power adapter	1
	Camera	1
 	Stationery card	A certain number

Requirements

- Take care when using electricity.
- Before the experiment, check whether the experimental equipment is complete and intact. If there is any omission or damage, please report to the teacher.
- Any specific operation in the experiment should be carried out according to the experiment manual. If you have any questions, please promptly ask the teacher.

- In the experiment, the joint of the robotic arm is in a working state after being powered on. Do not force it to move without pressing the unlock button.
- Equipment failure during the experiment must be reported to the teacher in time. Do not handle it by yourself.
- After the experiment, the equipment should be arranged well. You are allowed to leave the laboratory only after the inspection by the group leader.

Task 1: Training Model

Before a robot sorts stationery, we need to help it to know the stationery. Now, let us teach the robot how to do so.

1. Observations

Students observe how the teacher trains the sorting model, record key steps, and complete the blanks below.

Step 1: Collect some stationery photos.
Step 2: Add labels to them.
Step 3: ____
____

2. Steps

Step 1: Enable DobotScratch software.

Step 2: Select the Magician Lite device, and connect the "COM" port.

Step 3: Click the Add Extension button to add the AI module, as shown Figure 4.1.

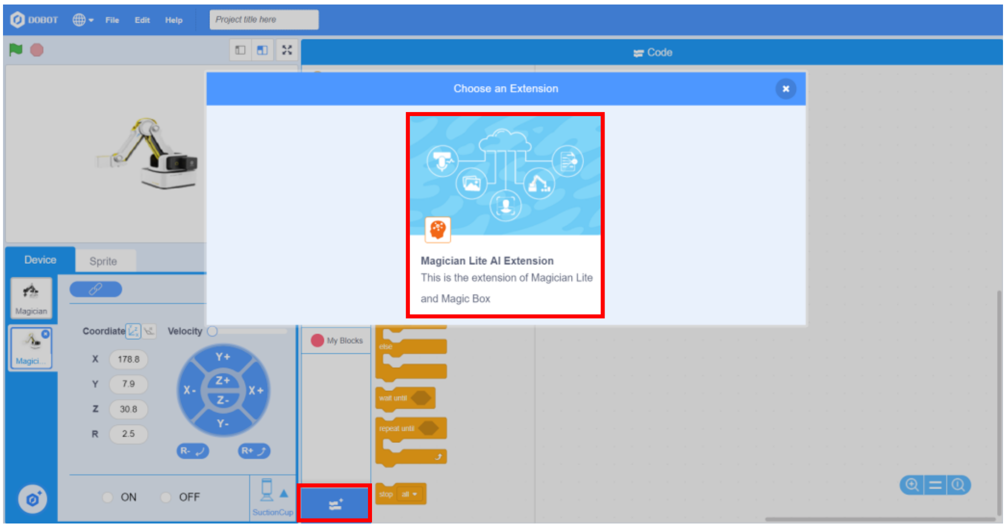


Figure 4.1 Adding the AI Extension module

Step 4: Click the AI tab, select the "New classification data" in the image recognition, and start to train the image classification model, as shown in Figure 4.2.

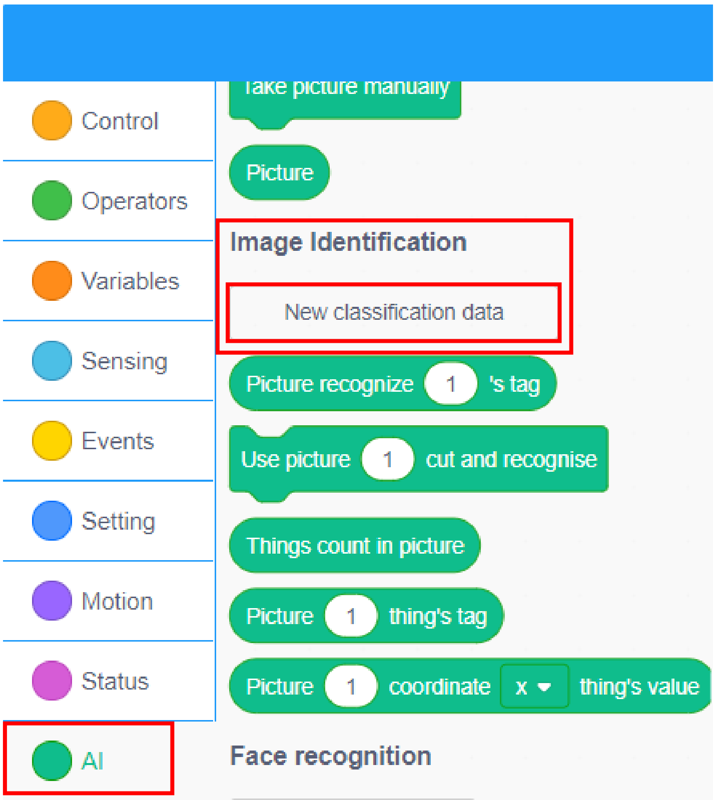


Figure 4.2 New classification data

Step 5: Select and start the camera on the robotic arms, as shown in Figure 4.3.

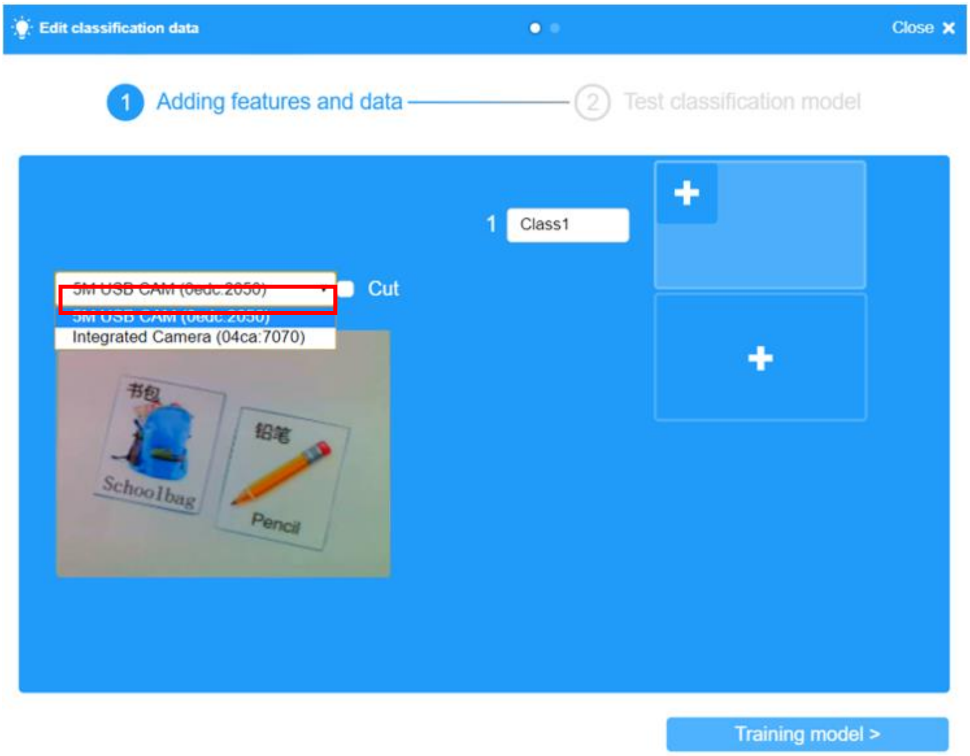


Figure 4.3 Starting the camera on the robotic arms

Step 6: Click the image segmentation icon, move the robotic arm by using the robotic arms control panel on DobotScratch, and adjust the camera position. Note that each stationery card is under the camera, and bears a yellow segmentation icon, as shown in Figure 4.4.

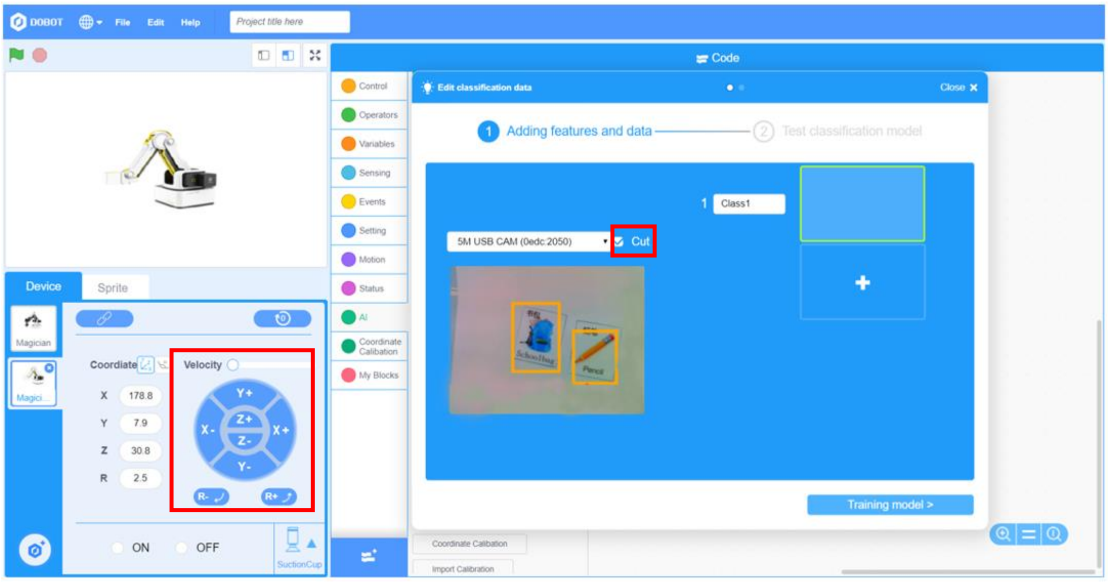


Figure 4.4 Adjusting the camera position

Step 7: Create a data set label. Click the plus sign to add a data set, and set the name of the data set to "Pencil" and "Schoolbag", as shown Figure 4.5.

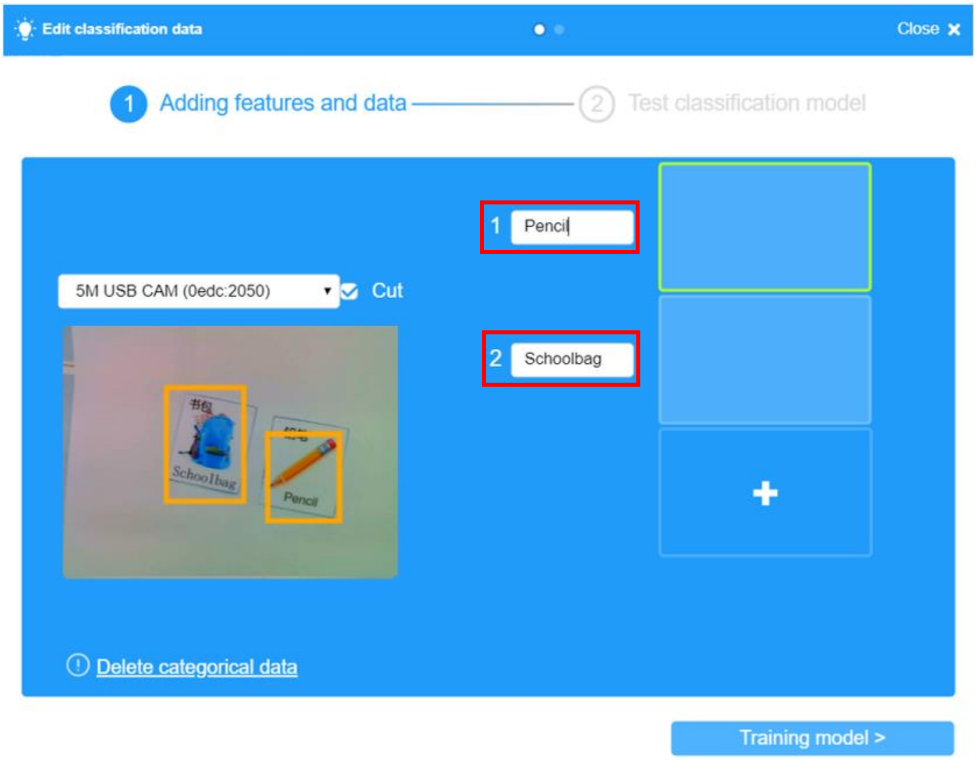


Figure 4.5 Creating the data set label

Step 8: Collect data. Click the segmented image. To make picture recognition more accurate, move the stationery card, and add the stationery cards at different angles, as shown Figure 4.6.

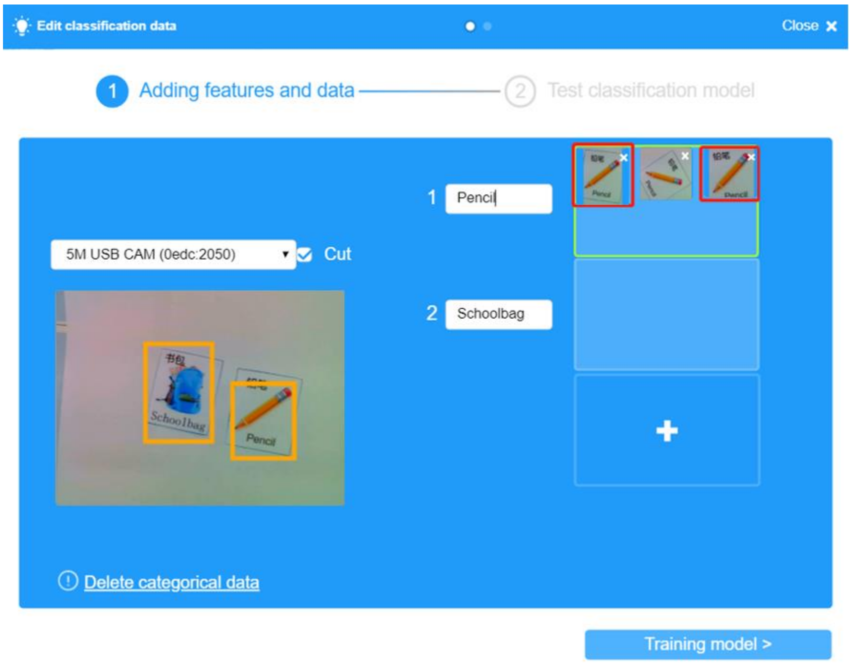


Figure 4.6 Collecting the image data

Step 9: Train the model. Click the "Training model" button, as shown in Figure 4.7.

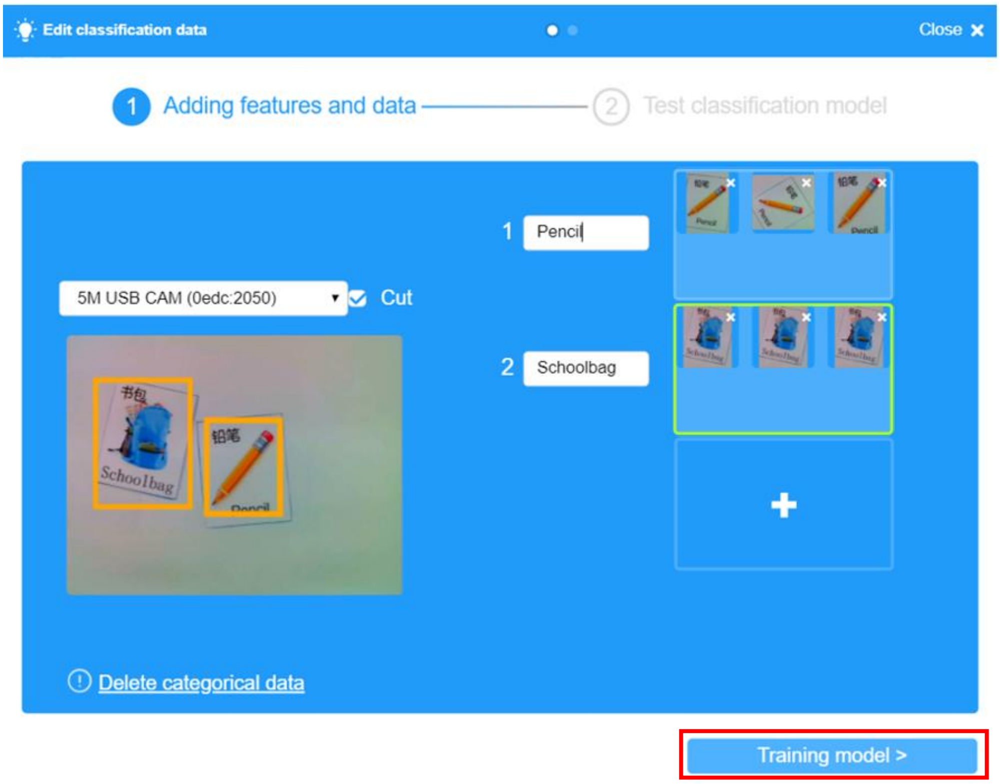


Figure 4.7 Training model

Step 10: Test the classification model. Click the camera on the robotic arms, and use it to get the test data, as shown in Figure 4.8.

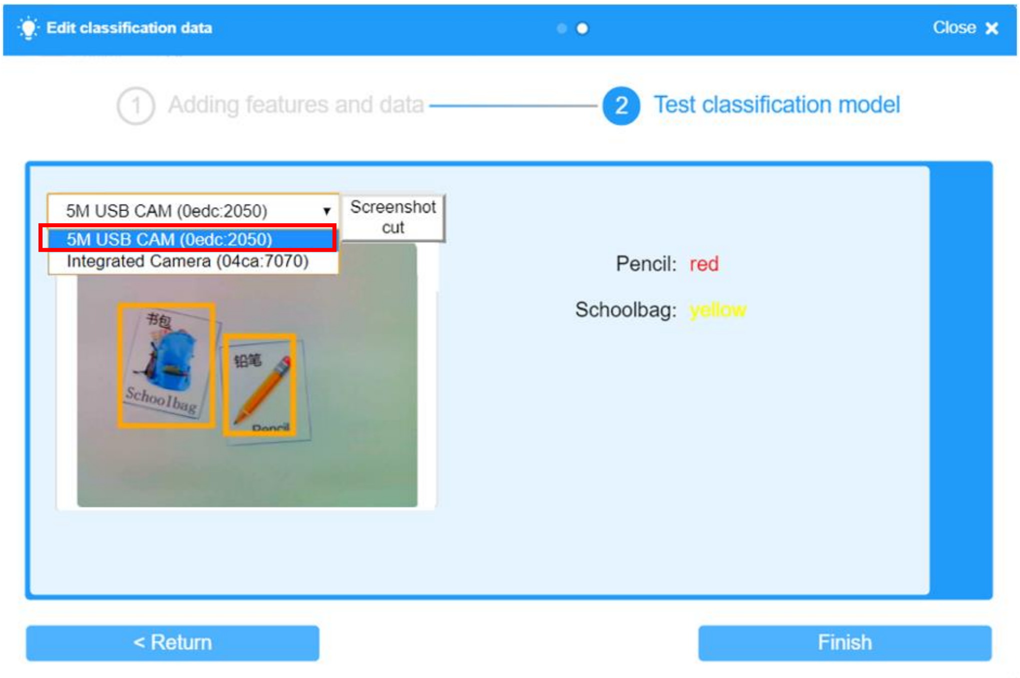


Figure 4.8 Getting the test picture from the camera

Step 11: Get the test result. Each label is provided with to a color, for example, the pencil corresponds to red, and the schoolbag to yellow. Click the "Screenshot cut" button to check whether the rectangle box color on each card is the same as that on the data set of the card. If yes, this indicates that the image classification model is created. Finally, click Finish, as shown in Figure 4.9.

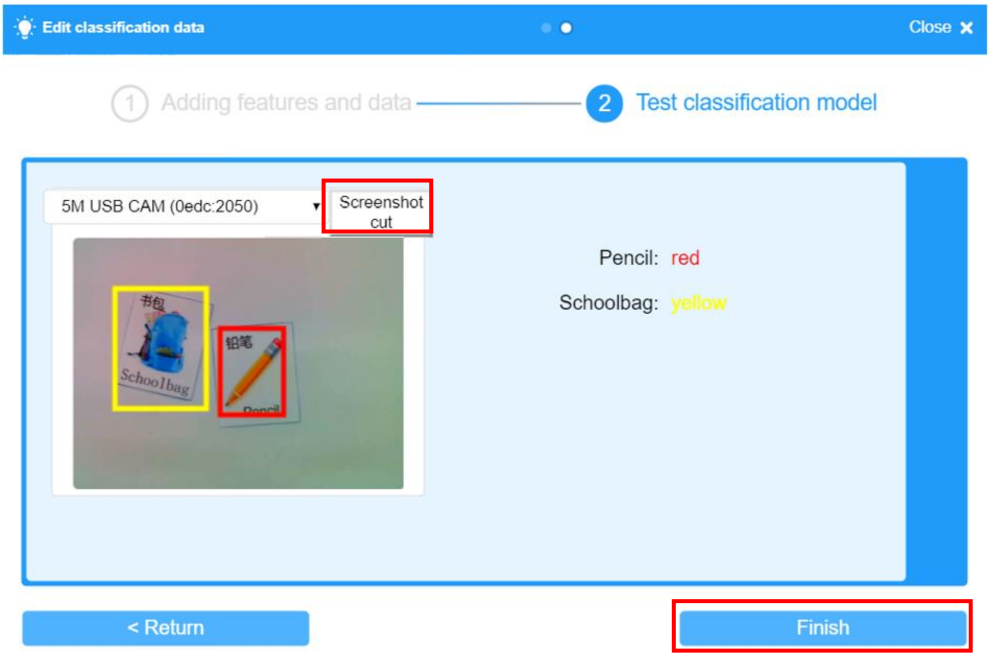


Figure 4.9 Checking the test result

Try training the classification model of three kinds of stationery.

3. Summary

（1）The steps for training the image classification model:__________

（2）After adding picture data, test the classification model in order to:

4. Self-Assessment

Check the completed contents in the experiment task. Tick the completed items, and circle the uncompleted items (◯).

Assessment Contents	Completion Status
I've collected the stationery picture data.	
I've learned how to add a label to picture data.	
I've completed the entire training of the image segmentation model.	

Task 2: Recognize & Segment Image

From the above task we have learned how to help a robot know a picture, that is, train the image segmentation model. Now we can apply a trained segmentation model to ensure that after segmenting an image, the robot recognizes its features.

1. Observations

Observe how the teacher segments and recognizes an image, record key experiment steps, and fill them in the blank.

Step 1: Get the desk photo.

Step 2: Segment and recognize the desk photo.

Step 3:____

____

2. Steps

(1) Prepare Hardware

Step 1: Design the experimental map, as shown in Figure 4.10, and set the equipment.

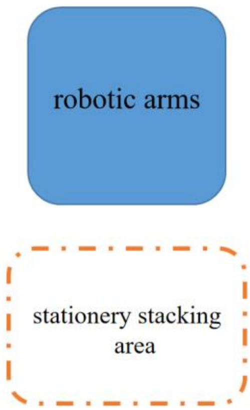


Figure 4.10 Image recognition and segmentation experiment

Step 2: Install a camera at the end of Magician Lite, as shown in Figure 4.11.

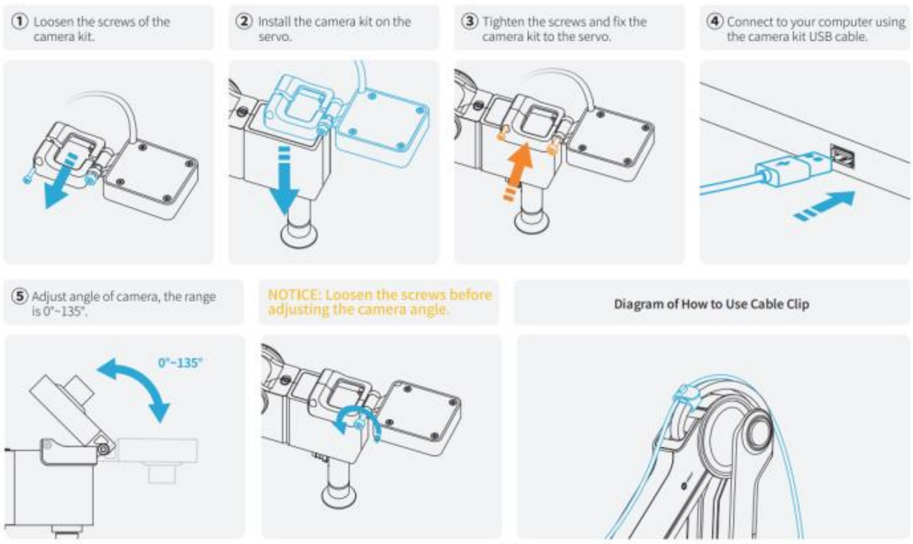


Figure 4.11 Installing the camera

Step 3: Start the robotic arm, and connect it to a computer.

(2) Design Program

Step 1: Analyze the program realization flow, and read the flow chart, as shown in Figure 4.12.

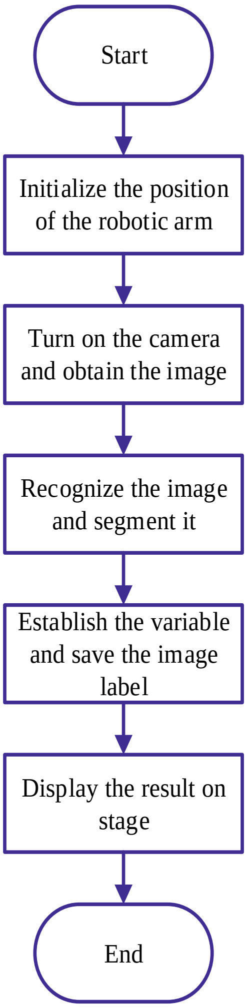


Figure 4.12 Image recognition and segmentation flow

Step 2: Initialize the position of the robotic arm. Control the robotic arm to a proper position, and place a stationery card within the photographing scope, as shown Figure 4.13.

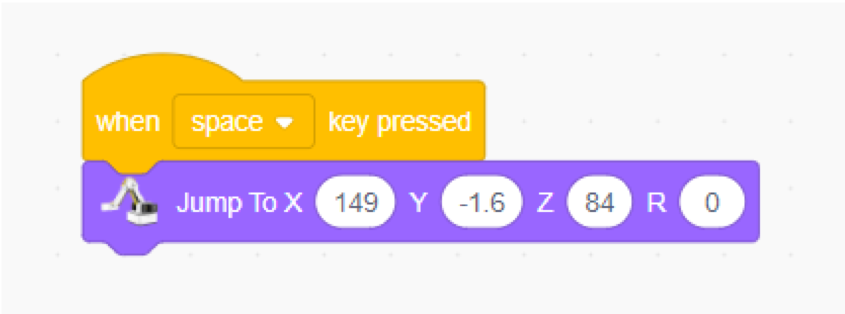


Figure 4.13 Initializing the position of the robotic arms

Step 3: Open the camera and get the picture. Click the "AI" tab, drag the block "Timeout (3)s to take picture" from "Image acquisition" to connect the previous block, as shown in Figure 4.14.

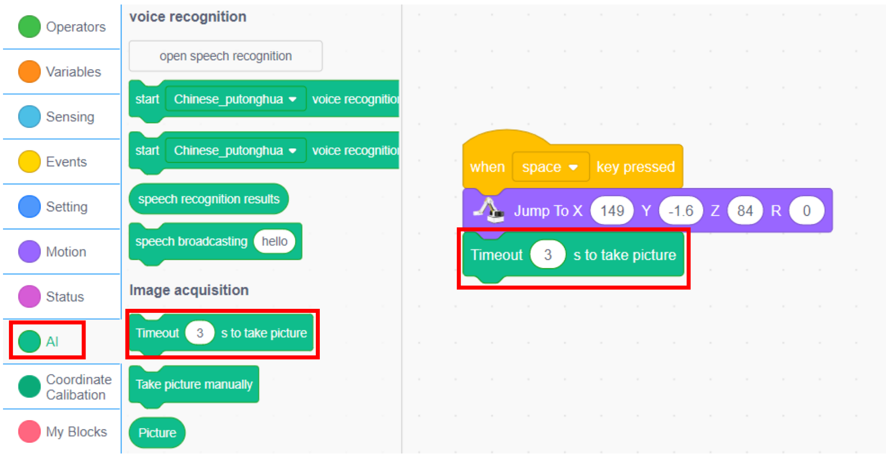


Figure 4.14 Getting the image

☛ The image acquisition module offers "Timeout (3)s to take picture" and "Take picture manually". Try using the two modules, and explain why the "Timeout (3)s to take picture" block is used.

Step 4: Segment and recognize the image. Click the "AI" tab, drag "Use picture (1) cut and recognise" from "image Identification", as shown in Figure 4.15.

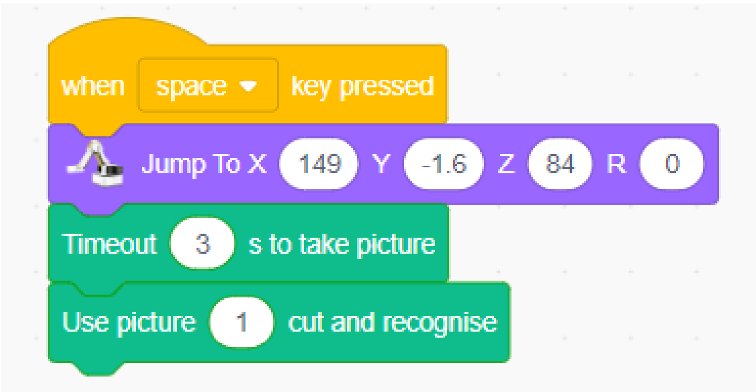


Figure 4.15 Segmenting and recognizing the image

Step 5: Set the taken image for segmentation and recognition, and piece "Picture" to the "1" position, as shown in Figure 4.16.

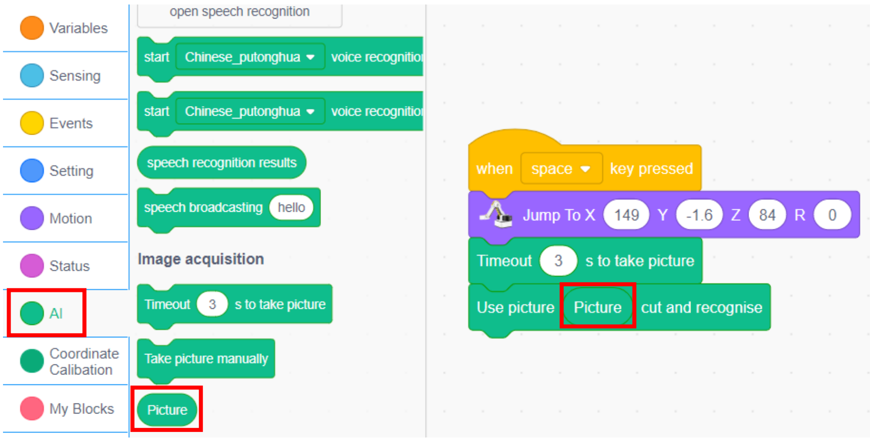


Figure 4.16 Setting the object for segmentation and recognition

Step 6: Create a variable, and name it *stationery name A* and *stationery name B*, as shown in Figure 4.17, as shown in Figure 4.17.

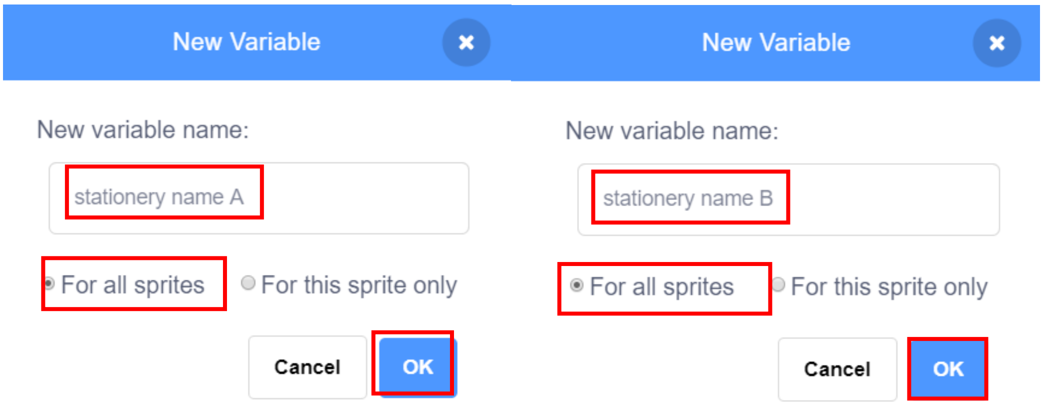


Figure 4.17 Creating the variable

What distinguishes "apply to all roles" from "apply to the current role"?

Step 7: Show the image label result on the stage, assign "Picture (1) thing’s tag" to *stationery name A*, and "Picture (2) thing’s tag" to *stationery name B*, as shown in Figure 4.18.

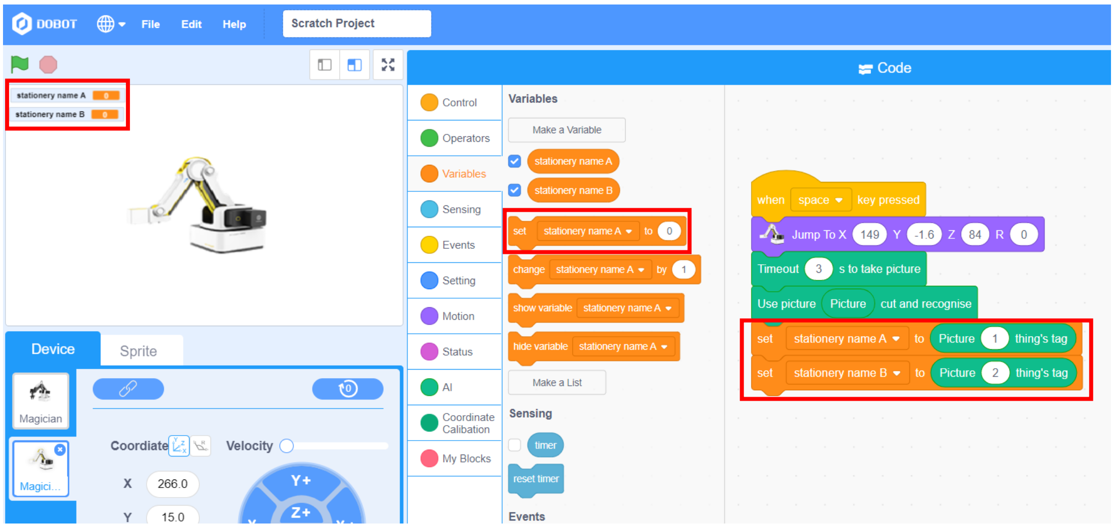


Figure 4.18 Assigning the variable

Step 8: Run the program, and check whether the image segmentation and recognition succeeds. Figure 4.19 shows the successful image segmentation and recognition, and Figure 4.20 shows the failed image segmentation and recognition.

Figure 4.19 Successful image segmentation and recognition

Figure 4.20 Failed image segmentation and recognition

Why image segmentation and recognition failed?

3. Summary

(1) The steps for creating a variable are:____________________________
____
____

(2) The key to successful image segmentation and recognition is:______
____

4. Self-Assessment

Check the completed contents in the experiment task. Tick the completed items, and circle the uncompleted items (◯).

Assessment Contents	Completion Status
I've used the photographing instruction.	
I've used the instruction of image segmentation and recognition.	
I've learned to define and assign the variable.	

Task 3: Complete Desk Arrangement

From the above task we learn to build the image segmentation model, test the image segmentation function and help the robot know stationery types. Now, we will arrange the desk with a robot, which will automatically set stationery on the desk to its proper position.

1. Observations

Observe how the teacher arranges a desk, record key experiment steps, and complete the blanks below.

Step 1: Take a photo of the stationery on the desk and get the image.

Step 2: Segment and recognize the picture.

Step 3: Test what the segmented picture shows. If it is a pencil, ________

____

If it is a schoolbag, ______________________________________

____

2. Steps

(1) Prepare Hardware

Step 1: Design the experiment map, and set the equipment. Figure 4.21 shows the reference map.

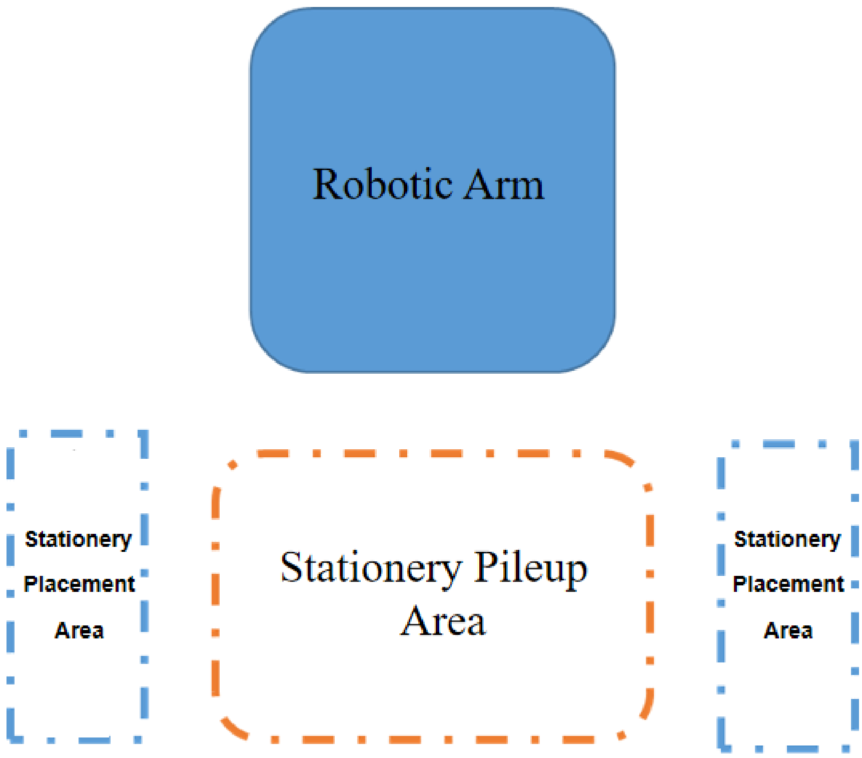


Figure 4.21 Stationery classification experiment

Step 2: Install the camera, connect it and the robotic arm to the computer.

(2) Design Program

Step 1: Analyze the stationery classification flow, and read the flow chart, as shown in Figure 4.22.

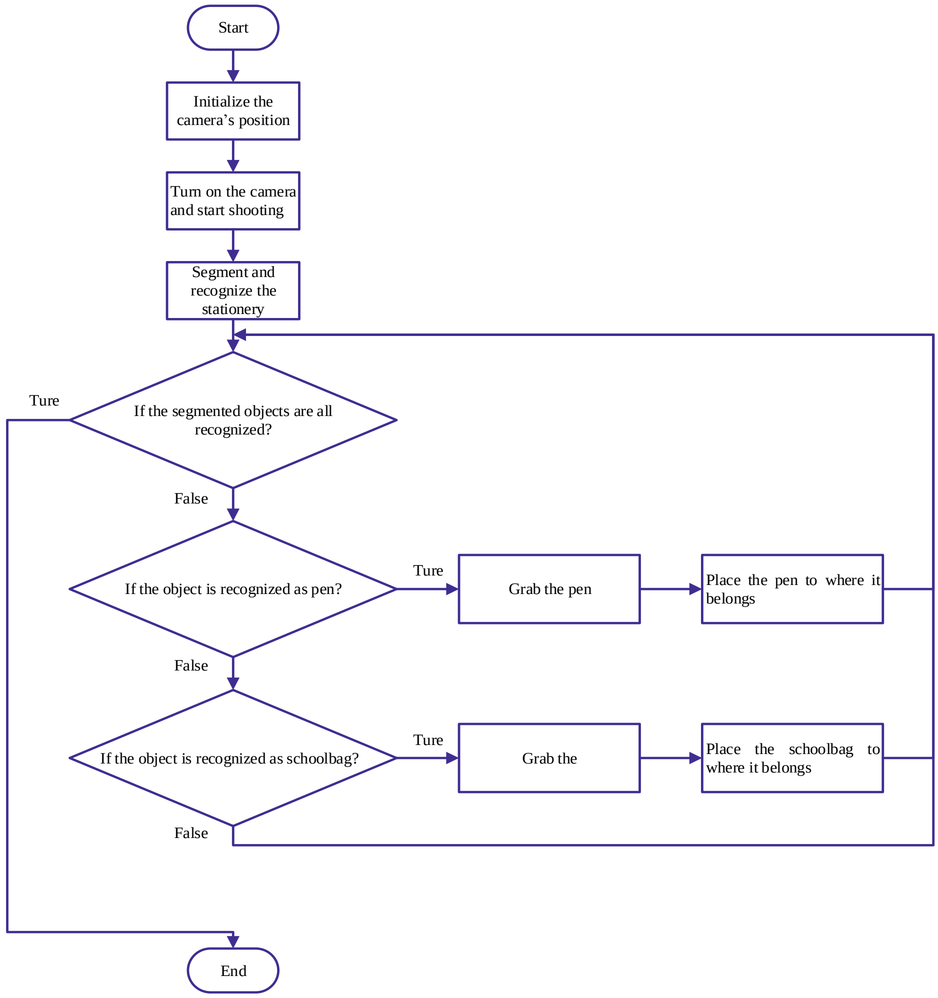


Figure 4.22 Stationery classification flow

Review task 2: steps for summarizing image segmentation and recognition

Step 2: Program image segmentation and recognition, as shown in Figure 4.23.

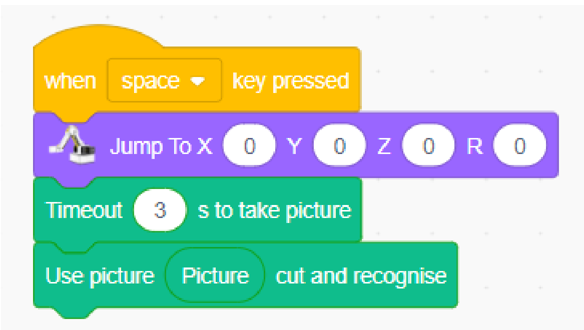


Figure 4.23 Image segmentation and recognition

Step 3: Define the variable *counter* to store the quantity of the items with image segmentation, as shown in Figure 4.24.

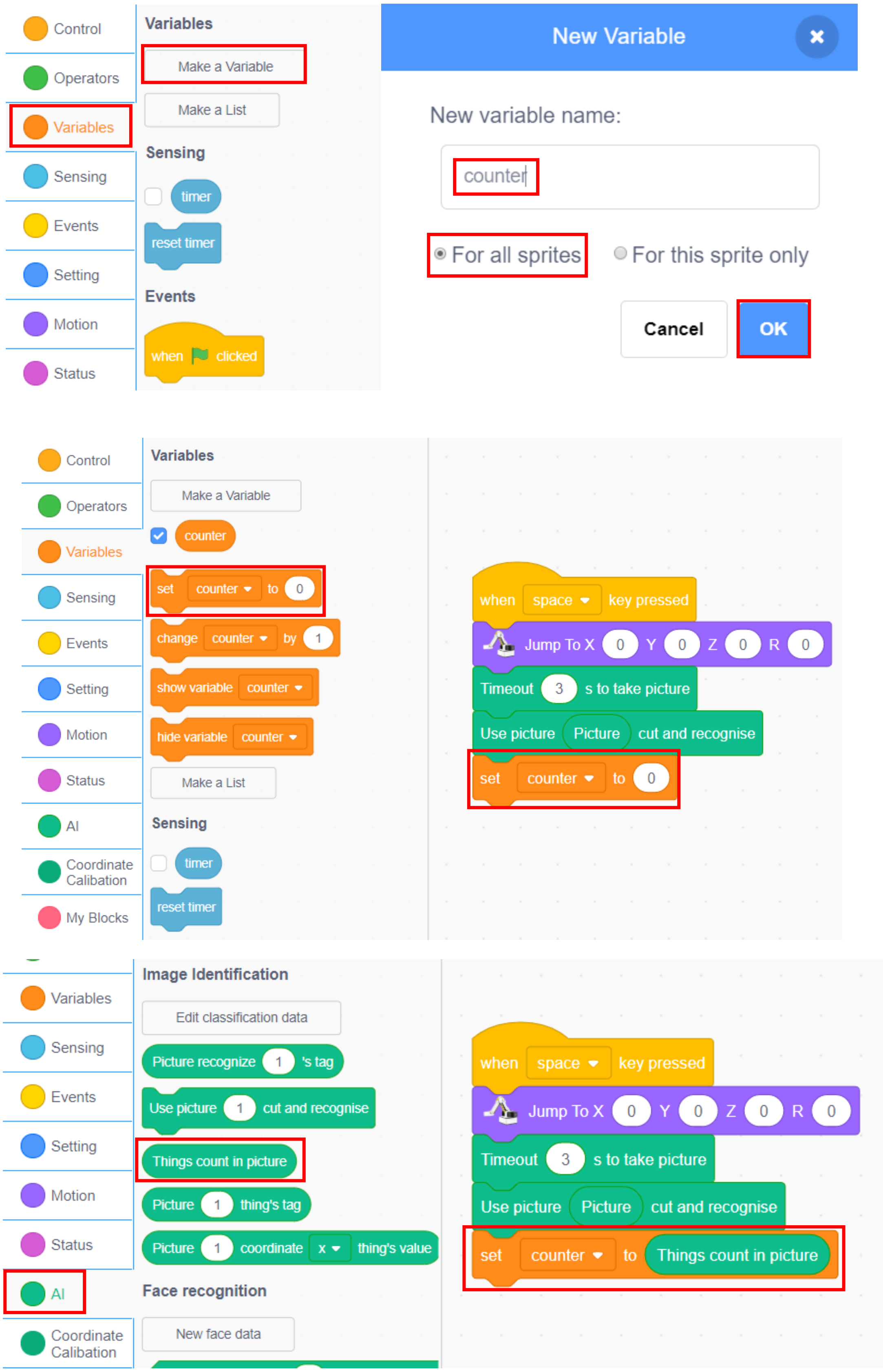


Figure 4.24 Defining the variable *counter*

Step 4: Define the variable *i* to store the sequence number of the items with image segmentation, and set the initial value to 1, as shown in Figure 4.25.

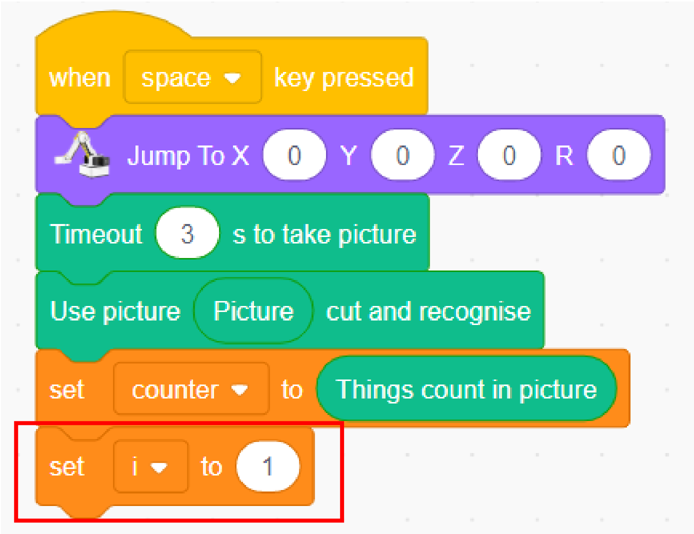


Figure 4.25 Defining the variable *i*

Step 5: Recognize the label of each item with image segmentation. If a label shows a pencil, the robotic arms will grab the pencil to the proper placement area, as shown in Figure 4.26.

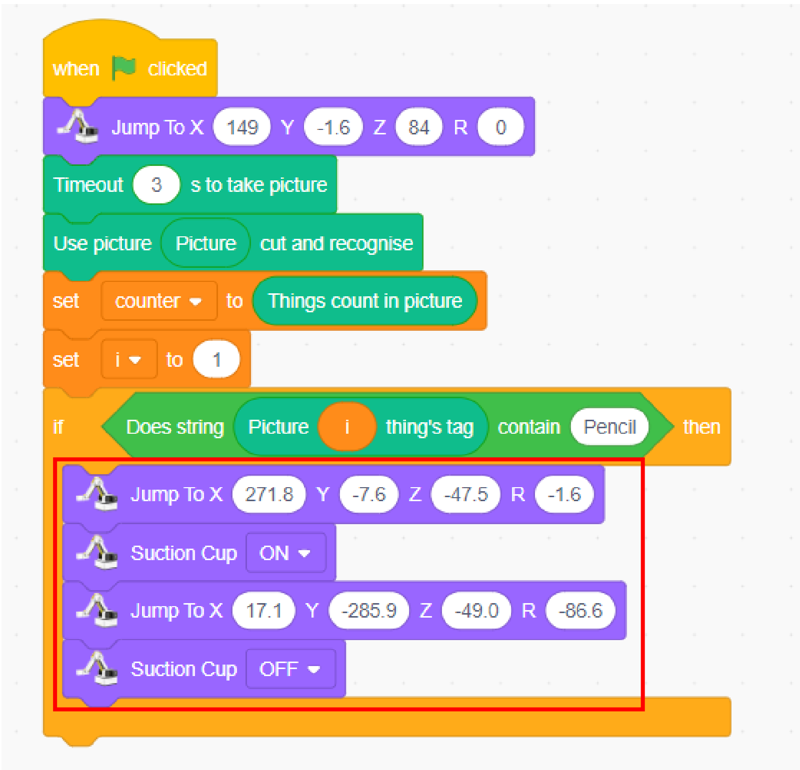


Figure 4.26 The pencil as a recognition result

Step 6: Make the robotic arms grab the schoolbag to the proper placement area if the label is recognized as schoolbag. The programming flowchart is as shown in Figure 4.27.

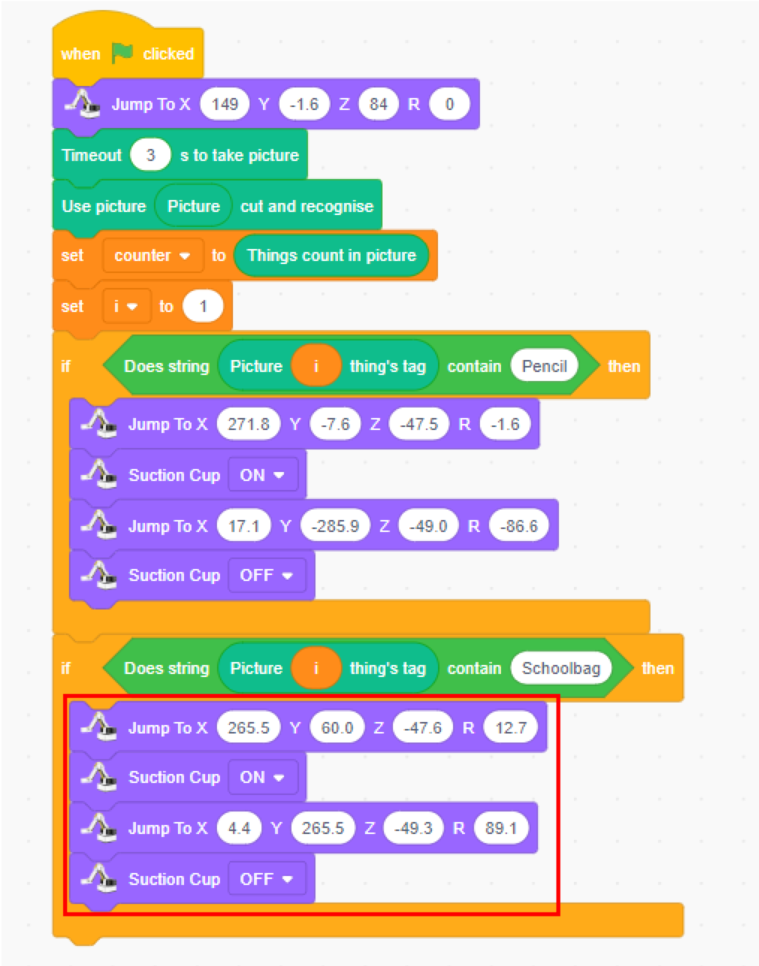


Figure 4.27 The schoolbag as a recognition result

Step 7: Repeat the above test process until all the segmented items are recognized, as shown in Figure 4.28.

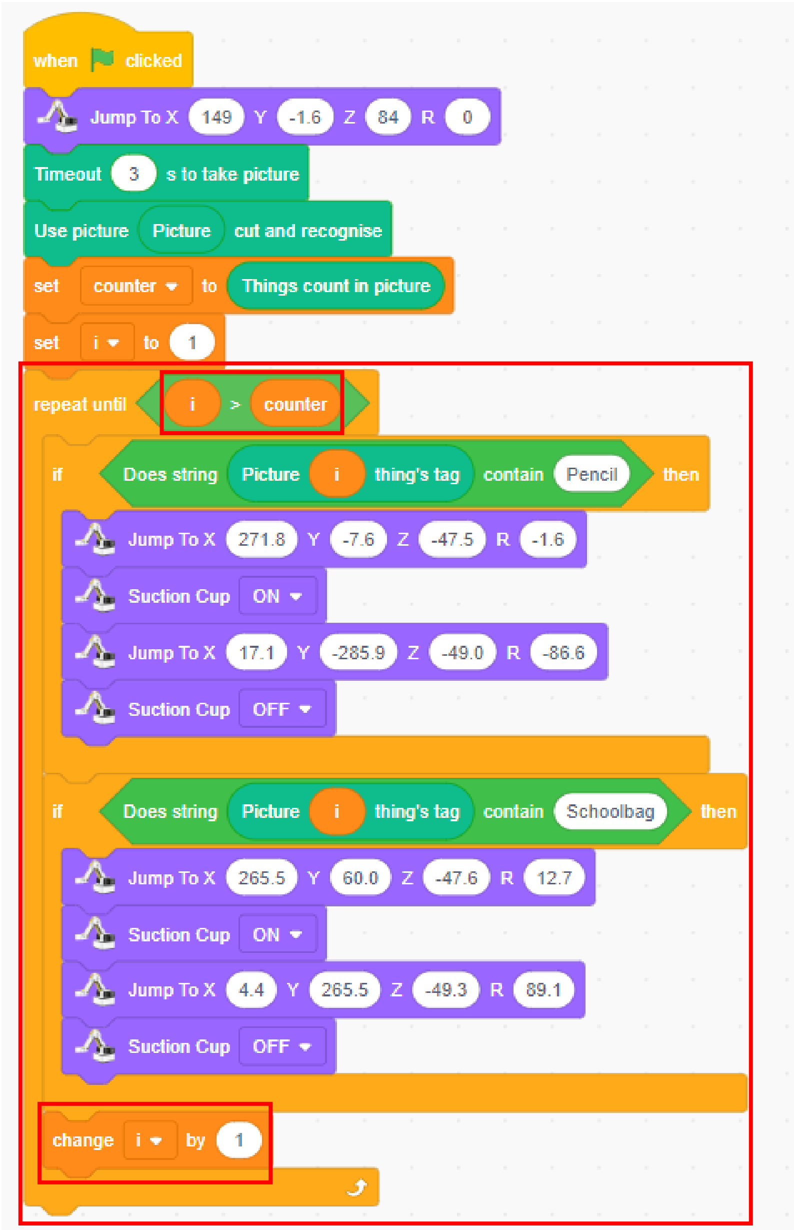


Figure 4.28 Repeating the execution

3. Summary

(4) The instruction of the character string recognition is:_____________

____

(5) The instruction of the conditional loop is:______________________

____

4. Self-Assessment

Check the completed contents in the experiment task. Tick the completed items, and circle the uncompleted items (◯).

Assessment Contents	Completion Status
I've learned to use the instruction of the character string test.	
I've learned to use the instruction of the conditional loop.	
I've programmed the desk arrangement.	

Task 4: Use the Multi-branch Structure

In the above task we have arranged a desk by using a single-branch structure. In the case of multiple judgment conditions, we use several single-branch structures. Thus, the computer can execute the judgment more times. To improve the program execution of the computer, we can replace multiple single-branch structures with a multi-branch structure. Now, let us use a multi-branch structure for programming.

1. Observations

Observe how the teacher writes a multi-branch structure program, record key steps and complete the blank below.

Step 1: Segment the items on the desk.
Step 2: Respectively test the segmented items.
Step 3: ____

____

2. Steps

(1) Prepare Hardware

Task 4 has the same hardware building as task 3. Students need to build the hardware in task 4 as per task 3.

(2) Design Program

Step 1: Analyze the realization flow of the desk arrangement program, and draw the flow chart as per the flow chart in task 3.

Step 2: Take a photo of the desk and get the photo, and segment and recognize it as per steps in task 3, as shown in Figure 4.29.

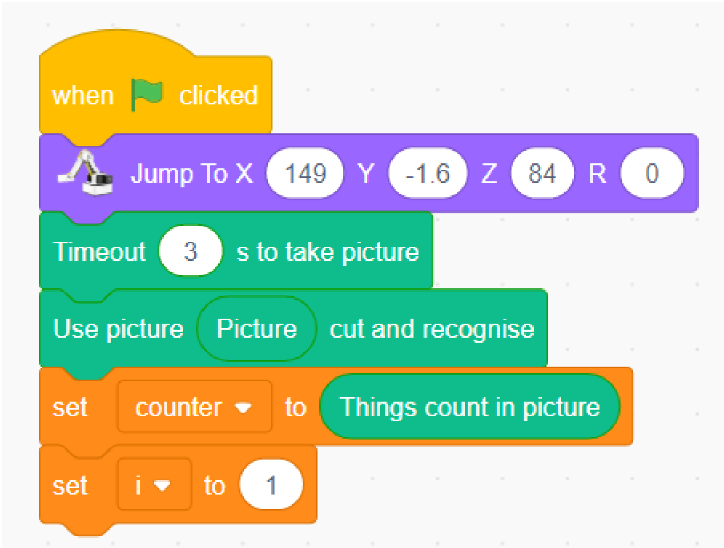
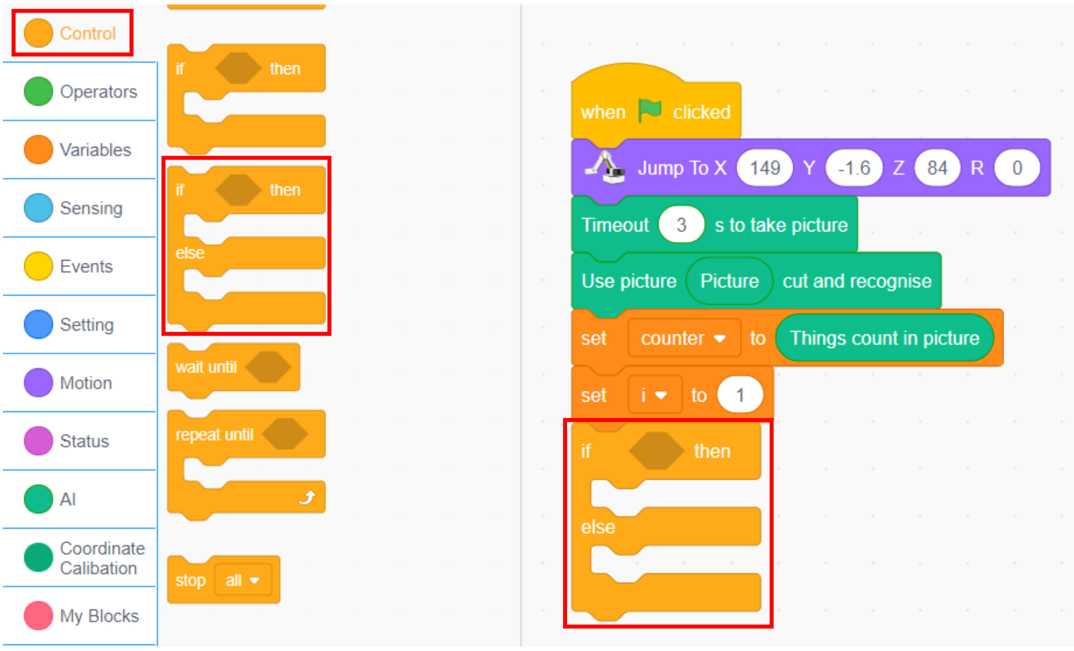


Figure 4.29 Image segmentation and recognition

Step 3: Test the label of a segmented item by using a multi-branch structure. If the label of a segmented item shows a pen, the robotic arm will grab the pencil to the proper placement area, as shown in Figure 4.30.



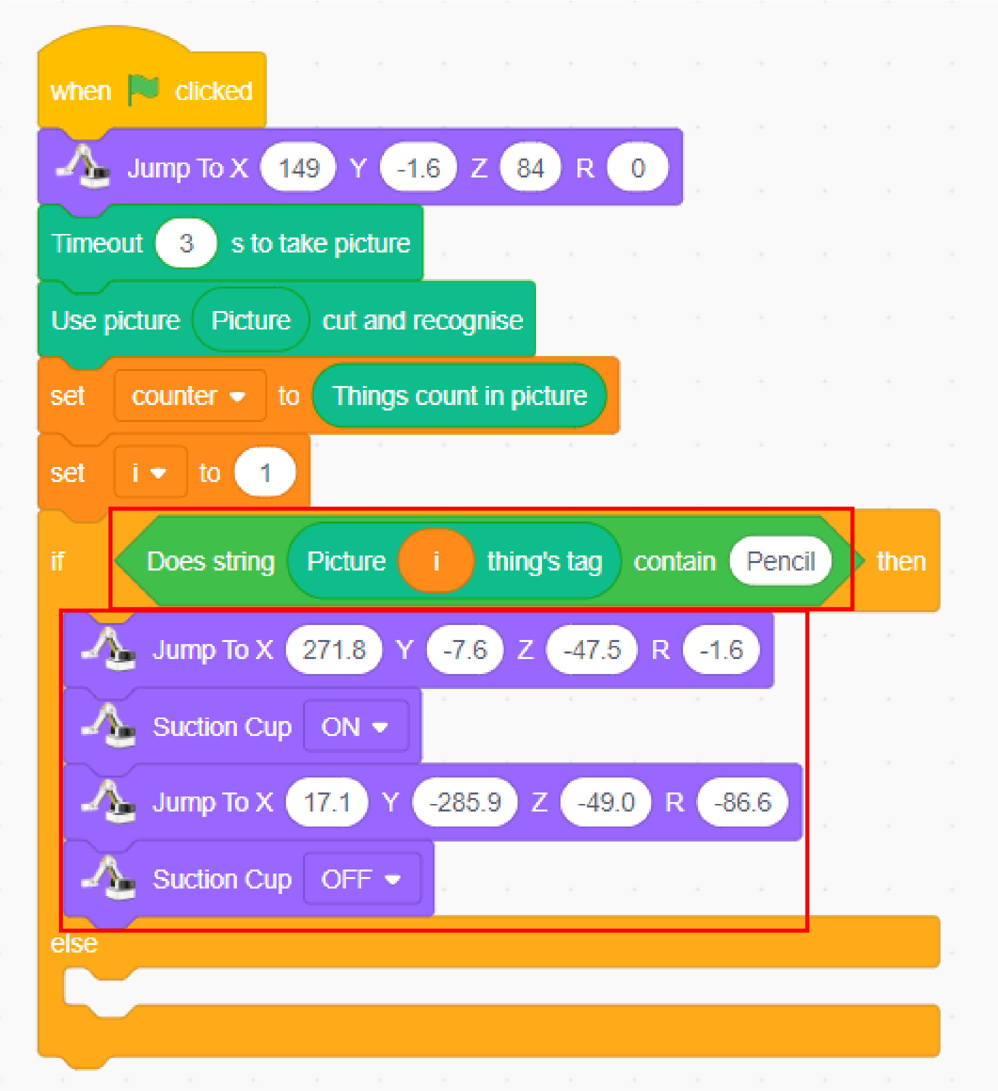


Figure 4.30 Multi-branch structure

Step 4: Make the robotic arm place the schoolbag to where it belongs if the label is recognized as schoolbag, as shown in Figure 4.31.

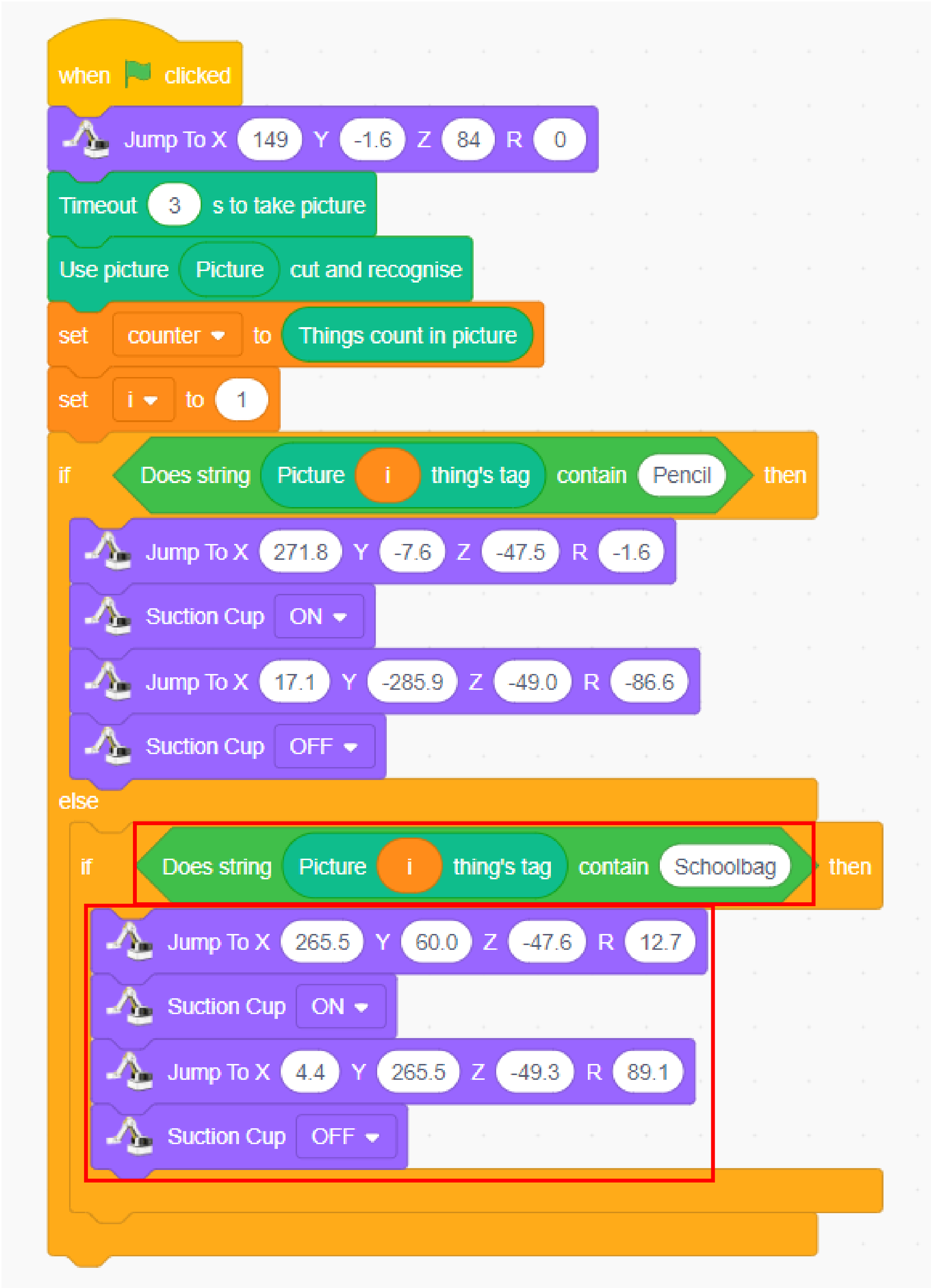


Figure 4.31 Placing the schoolbag

Step 5: Repeat the above test process until all the segmented items are recognized, as shown in Figure 4.32.

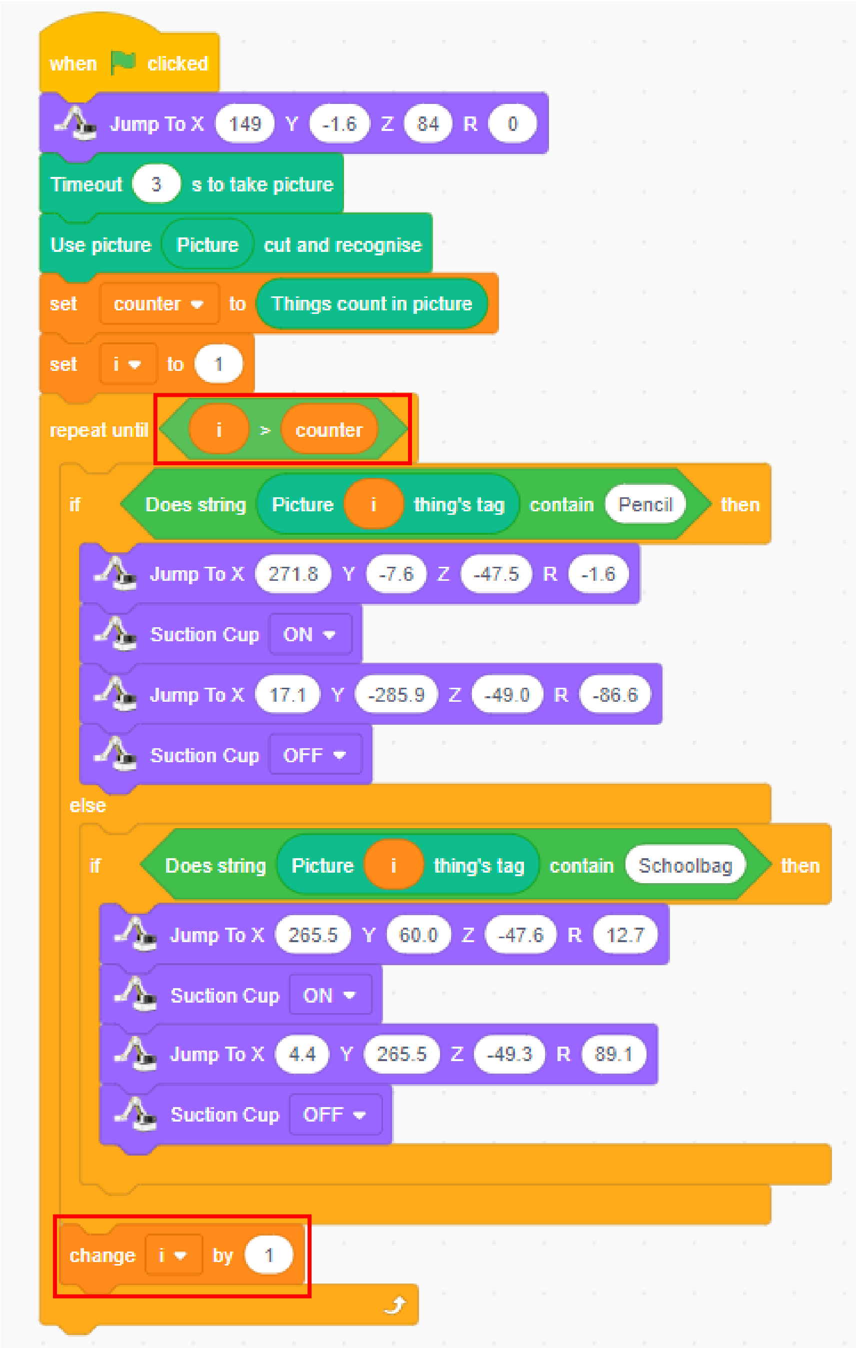


Figure 4.32 Repeating the execution

Program the arrangement of three pieces or more of stationery on the desk by yourself.

3. Summary

(1) The instruction of the multi-branch structure is:____________________

____

(2) The difference between a multi-branch structure and multiple single-branch structures is:_____________________________________

____

4. Self-Assessment

Check the completed contents in the experiment task. Tick the completed items, and circle the uncompleted items (◯).

Assessment Contents	Completion Status
I've learned to use a multi-branch structure.	
I've known the difference between a multi-branch structure and multiple single-branch structures.	

Experiment 5

Smart Cafeteria

Team Name: Team member: Date:

Overview

What do you think the smart cafeteria should look like? The smart cafeteria has modules such as smart ordering and cafeteria management, and each module has multiple functions, which can improve the dining satisfaction. This experiment uses speech recognition technology and facial recognition technology to simulate voice ordering and facial recognition payment in the smart cafeteria. Among them, the voice ordering allows the robotic arm to automatically take meals according to the speech recognition result. Facial recognition payment allows the system to verify identities through facial recognition and deduct the corresponding amount from the account. Let's experience it together!

Objectives

1. Master the usage of speech recognition instructions by invoking them.
2. By completing the tasks of facial data collection and training, understand the process of facial data collection and training.
3. Master the use of judgment structure by writing a program for voice ordering.
4. Master the use of facial recognition instructions by writing a facial recognition payment program.
5. Understand how to use variables by writing a facial recognition payment program.

Equipment

Icon	Name	Quantity

	Dobot Magician Lite robotic arm	1
	Suction cup kit	1
	Camera kit	1
	Type-C cable	1
	Power adapter	1
	Food stickers	1

Experiment Requirements

- Take care when using electricity.
- Before the experiment, check whether the experiment equipment is complete and intact. If there is any omission or damage, please report to the teacher.
- Any specific operation in the experiment should be carried out according to the experiment manual. If you have any questions, please promptly ask the teacher.

- In the experiment, the joint of the robotic arm is in a working state after being powered on. Do not force it to move without pressing the unlock button.
- Equipment failure during the experiment must be reported to the instructor in time. Do not handle it by yourself.
- After the experiment, the equipment should be arranged well. You are allowed to leave the laboratory only after the inspection by the group leader.

Task 1: Invoke Speech Recognition Instruction

Automatically collect voice and display the speech recognition result on the stage through speech recognition instruction.

1. Observations

Please complete the experiment steps according to the prompts.

Add "AI" extension module > Select speech recognition module > Create a variable >________________________________

____

________________________(Follow the prompts to complete the remaining steps)

2. Steps

Step 1: Add the "AI" extension module, click "**open speech recognition**" at the speech recognition section, and test the speech recognition results to ensure that the voice can be collected and recognized normally, as shown in Figure 5.1.

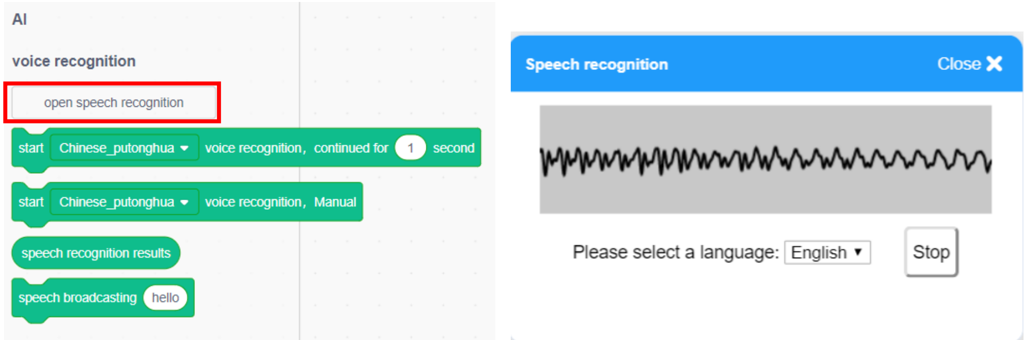


Figure 5.1 Opening the recognition window

Step 2: Drag out the "start English voice recognition, continued for 1 second" block, and change 1 second to 6 seconds, as shown in Figure 5.2.

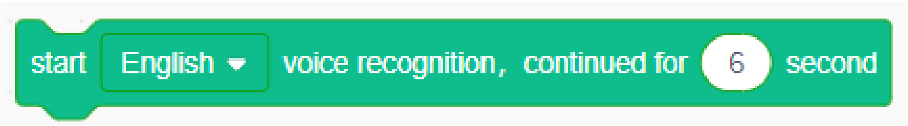


Figure 5.2 "start English voice recognition, continued for 1 second" blocks

Step 3: Click the "**Variables**" tab, and select "Create a variable" to create a variable "i", as shown in Figure 5.3.

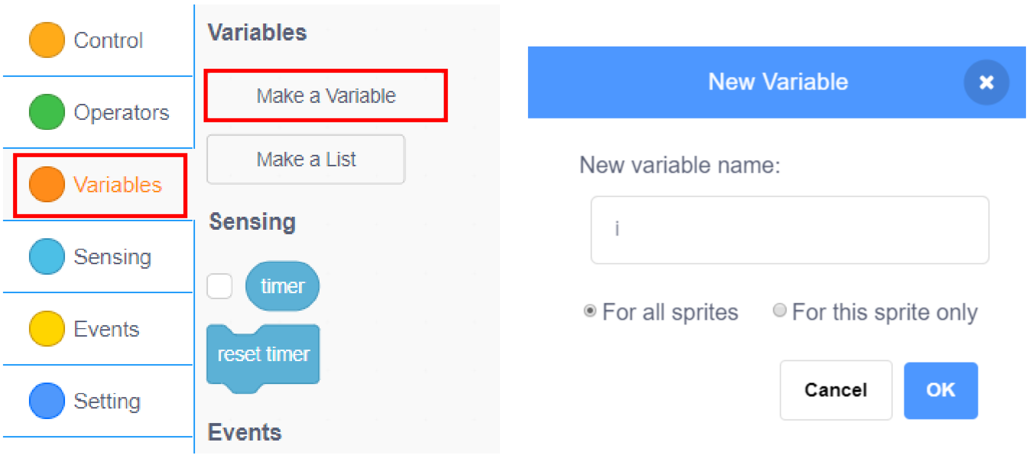


Figure 5.3 Creating a variable

Step 4: Click the "**Variables**" tab, drag the "Set i to (0)" block, click the "AI" tab, drag the "speech recognition results" block, and set the value of "i" to "speech recognition results", as shown in Figure 5.4.

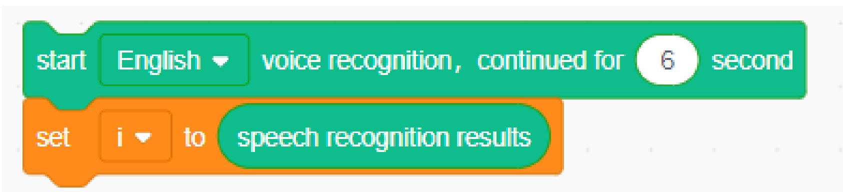
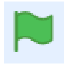


Figure 5.4 Setting the variable "i" to "speech recognition results"

Step 5: Click the "**Events**" tab and drag the "When  is clicked" block to trigger the program to run as shown in Figure 5.5.

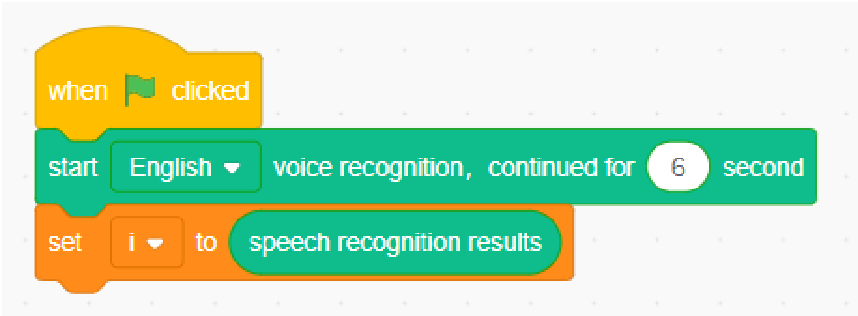


Figure 5.5 "When the green flag is clicked" triggers the program to run

Step 6: Click the small green flag to record the sound. You can see the speech recognition results on the stage. As shown in Figure 5.6.

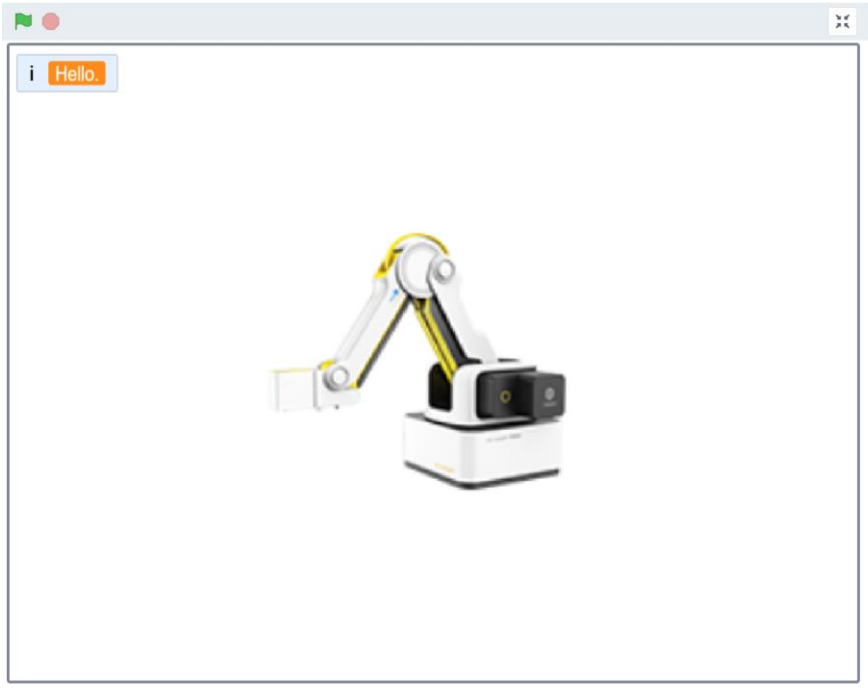


Figure 5.6 Speech recognition results displayed on the stage

Replace the "start English voice recognition, continued for 1 second" block with "start English voice recognition, Manual", and run the program. Observe what is the difference?

3. Summary

Summarize the steps to display the speech recognition results on the stage: ____
____
____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I've completed the task of creating a variable	
I've completed the task of displaying speech recognition results on stage	

Task 2: Collect & Train Facial Data

To perform facial data collection and training, add three persons' tags and collect pictures corresponding to three persons', train face models, and test face classification models.

1. Observations

Please observe how the teacher collect and train facial data.

Facial data creation: add facial features and data > ___________
____
____
(Follow the prompts to complete the remaining steps)

2. Steps

Step 1: Create facial data, as shown in Figure 5.7.

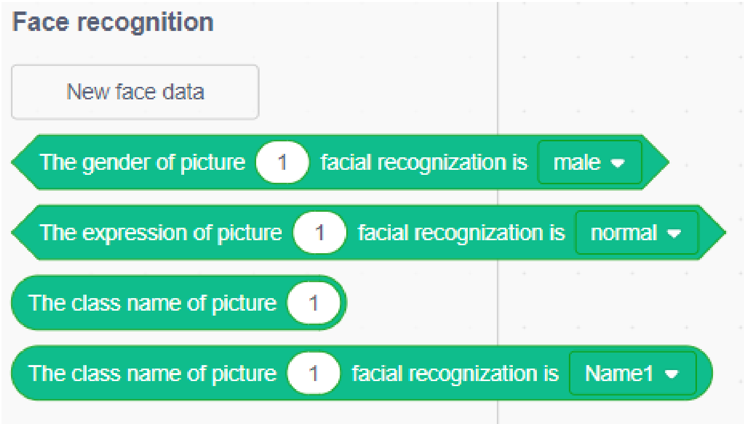


Figure 5.7 Creating facial data

Step 2: Click the "**New face data**" button to collect facial images and complete the facial data storage, as shown in Figure 5.8.

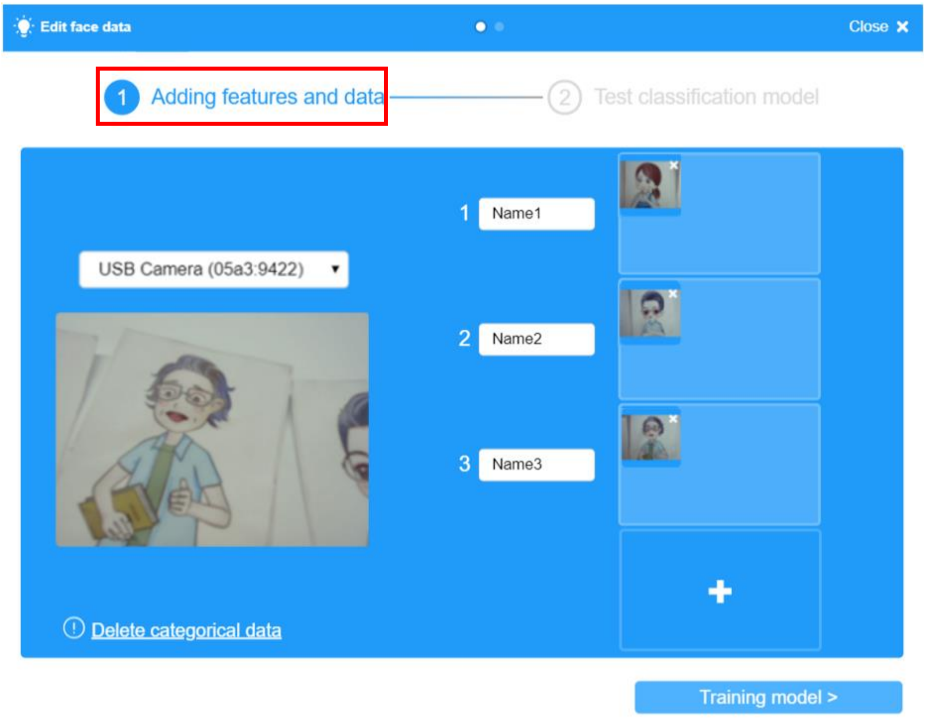


Figure 5.8 Adding features and data

Step 3: Train the model. The process of facial image preprocessing and feature extraction is the process of computer learning. Figure 5.9 shows the model training.

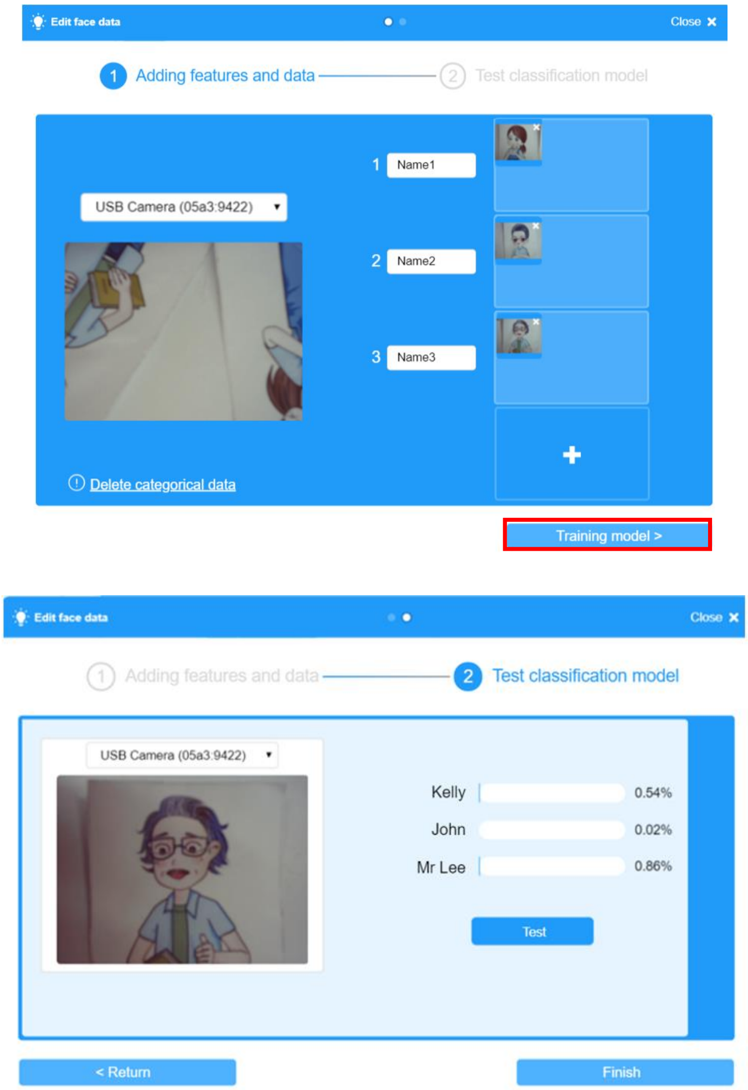


Figure 5.9 Training the model

Step 4: Test the classification model. Model testing is the process of facial image recognition and classification. Figure 5.10 shows the testing.

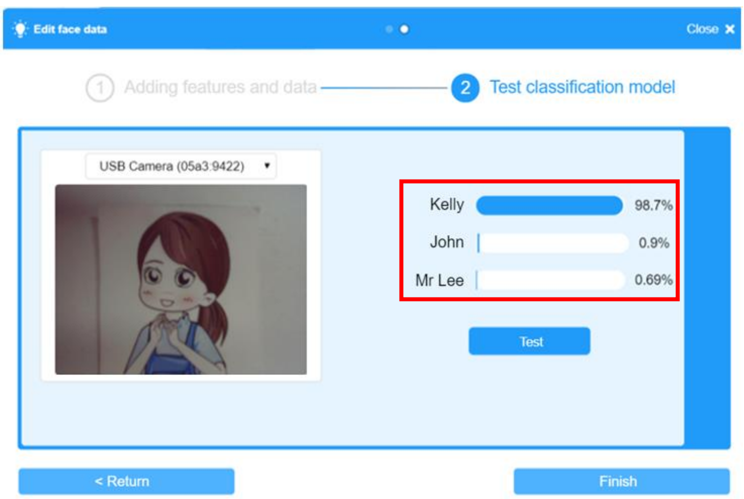


Figure 5.10 Testing

👉 Implement the following through programing: If the facial recognition result is "Kelly", then the voice "Kelly, hello!" is played.

3. Summary

Summarize the steps of facial data collection and training: ________
____
____
____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I’ve completed the task of facial data collection	
I’ve completed the task of testing the face classification model	
I’ve completed the questions raised in the experiment steps	

Task 3: Order by Voice

After recognizing what the orderer says, the robotic arm performs the food fetching action. For example, if John said "I want meal package A", the keyword “Meal package A" is recognized, and then the robotic arm is driven to get the Meal package A.

1. Observations

Please observe how the voice ordering system works and complete the steps of designing program based on your observation and prompts.

Program design: Initialize > collect voice > recognize the speech > If the speech recognition is “Meal package A" > make the robotic arm get the meal package A and sets the set meal price. ____

____

____

(Follow the prompts to complete the remaining steps)

2. Steps

(1) Prepare Hardware

Step 1: Prepare cards for meal packages A, B, C, and D.

Step 2: Power on the robotic arm, connect the robotic arm to the personal computer (PC), and install the end of the suction cup.

Step 3: Place the equipment, as shown in Figure 5.11.

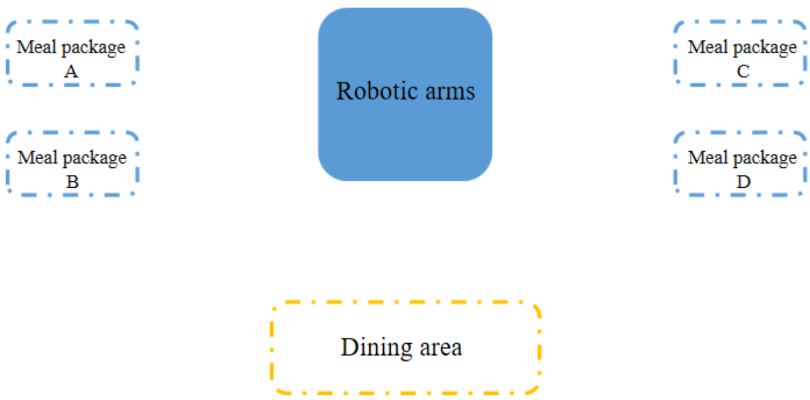


Figure 5.11 Placing the equipment

(2) Design Program

Step 1: Analyze the voice ordering process and draw a flowchart, as shown in Figure 5.12.

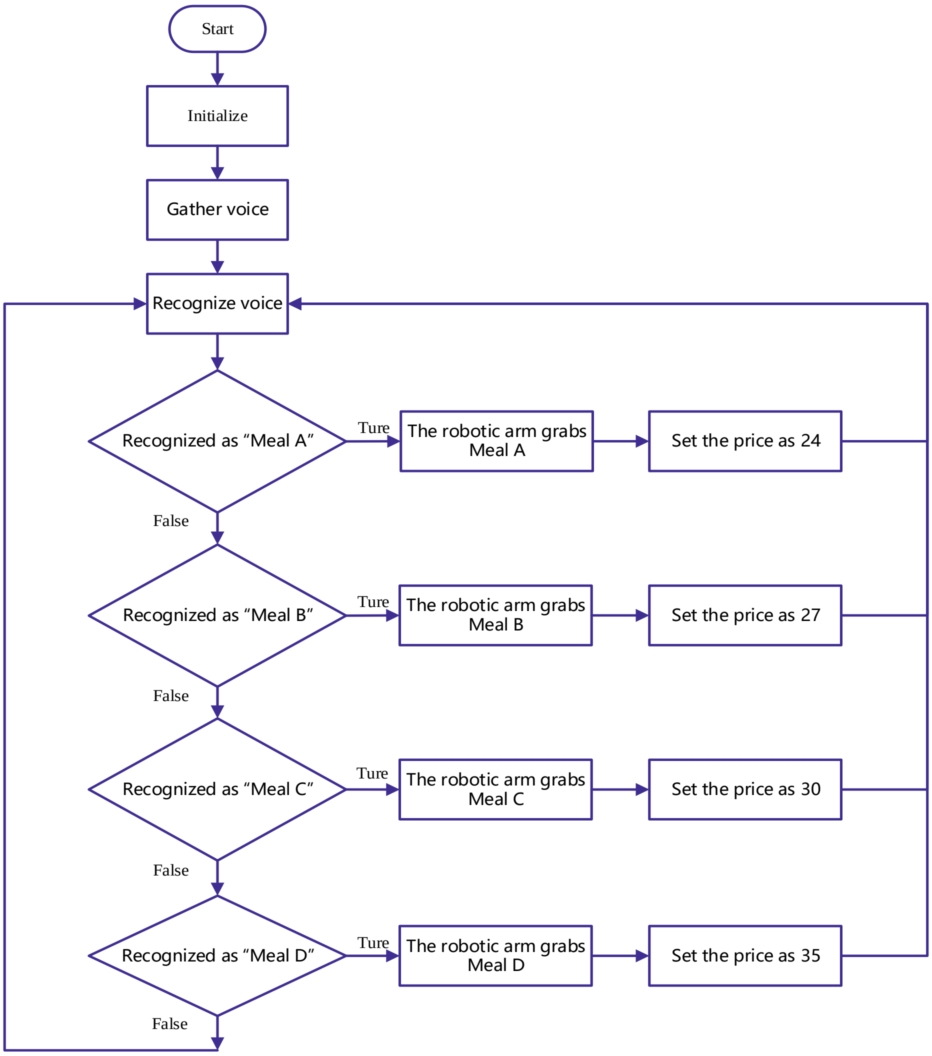


Figure 5.12 Flowchart

Step 2: Open the software, select the device Magician Lite, connect to the device, and add the "AI" extension module.

Step 3: Click the "**Events**" tab, drag the "When space key pressed" block, click the "**Setting**" tab, drag the "Select End Effector Suction Cup" block, and then put them together, such as Figure 5.13.

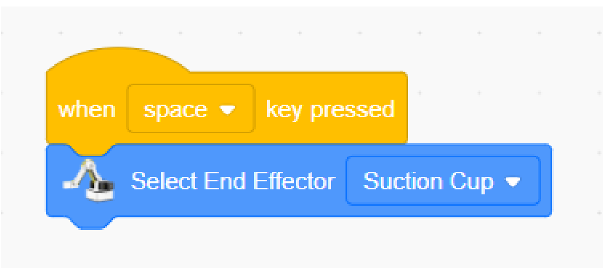
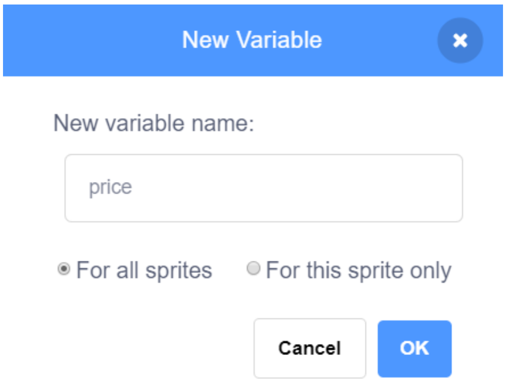


Figure 4.13 Setting the program starting manner and the end tool

Step 4: Click the "**Variables**" tab, select "Create a variable", and name it "price", as shown in Figure 5.14.



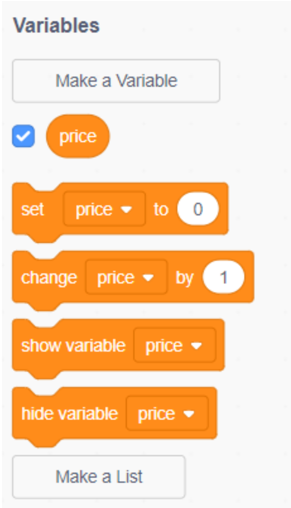


Figure 5.14 Create a variable "price"

Step 5: Set the variable price to "0" to initialize the end position of the robotic arm, as shown in Figure 5.15.

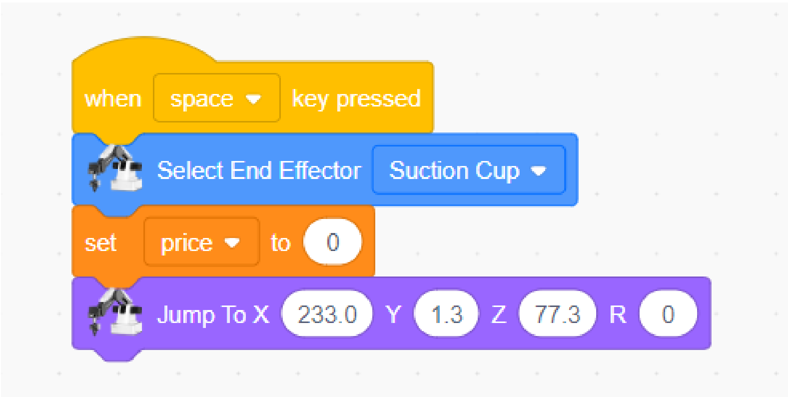


Figure 4.15 Initialization

Step 6: Click the "**AI**" tab and drag the "start English voice recognition, continued for 6 second" block to collect voice and recognize speech, as shown in Figure 5.16.

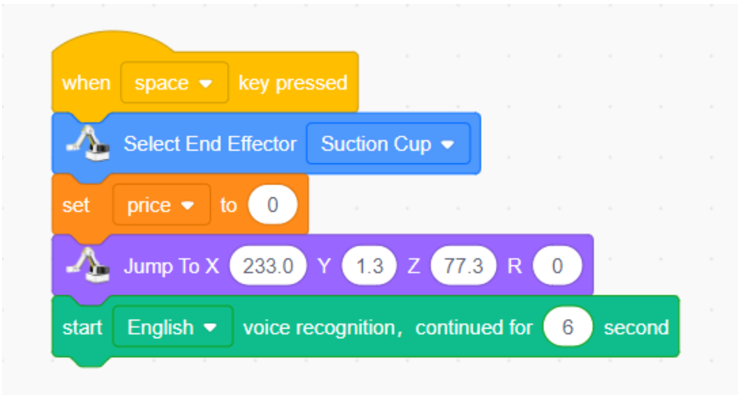


Figure 5.16 voice collection and speech recognition

Step 7: Determine the result of speech recognition. If it is “meal package A", play the voice "OK, please wait a moment"; the robotic arm gets the meal package A; the variable "price" is set to 24, which is the price of the meal package A; play the voice "This is your meal package A". Among them, the "Does string ... contain ..." block is in the calculation tab page; the "wait 1 seconds" and "if ... then ... else ..." block is in the control tab page; "speech recognition results" and "speech broadcasting" blocks are in the AI tab page. Drag the blocks to the code editing area, and put them together according to a certain logic, as shown in Figure 5.17.

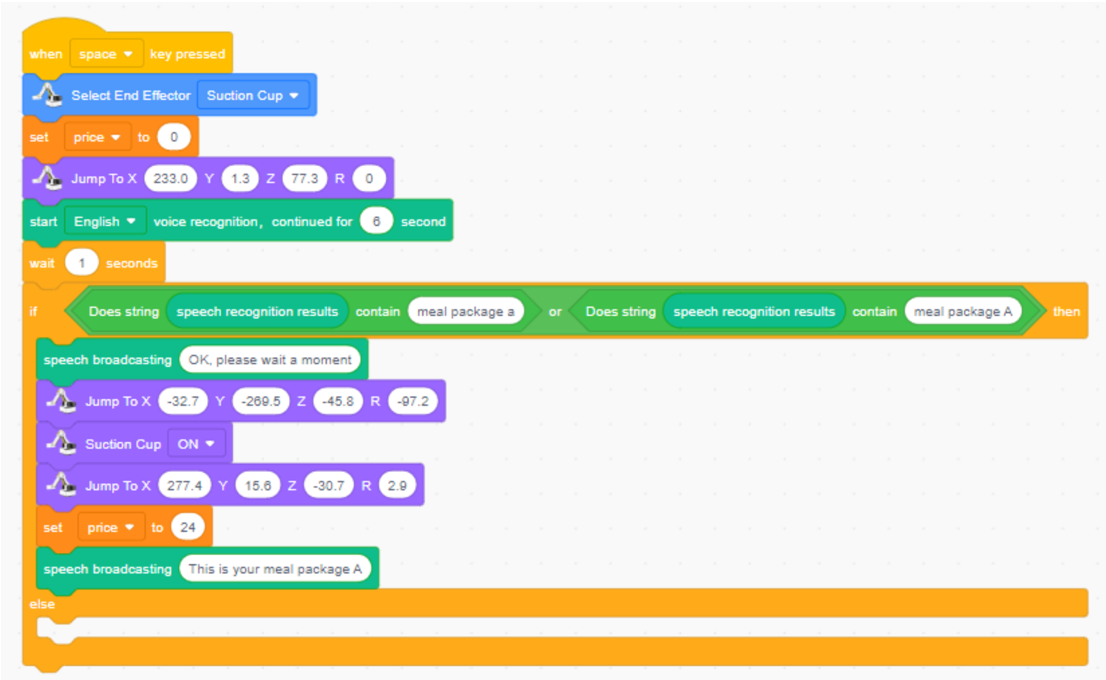


Figure 5.17 Speech recognition food fetching

Step 8: Determine whether the speech recognition result is “meal package B, meal package C, meal package D" according to the same logic. As the entire process is not executed only once, but multiple times, select "forever" on the control tab page. "The parts that need to be repeatedly executed are put into the repeating executing blocks. Here meal packages A and B are used as an example, as shown in Figure 5.18.

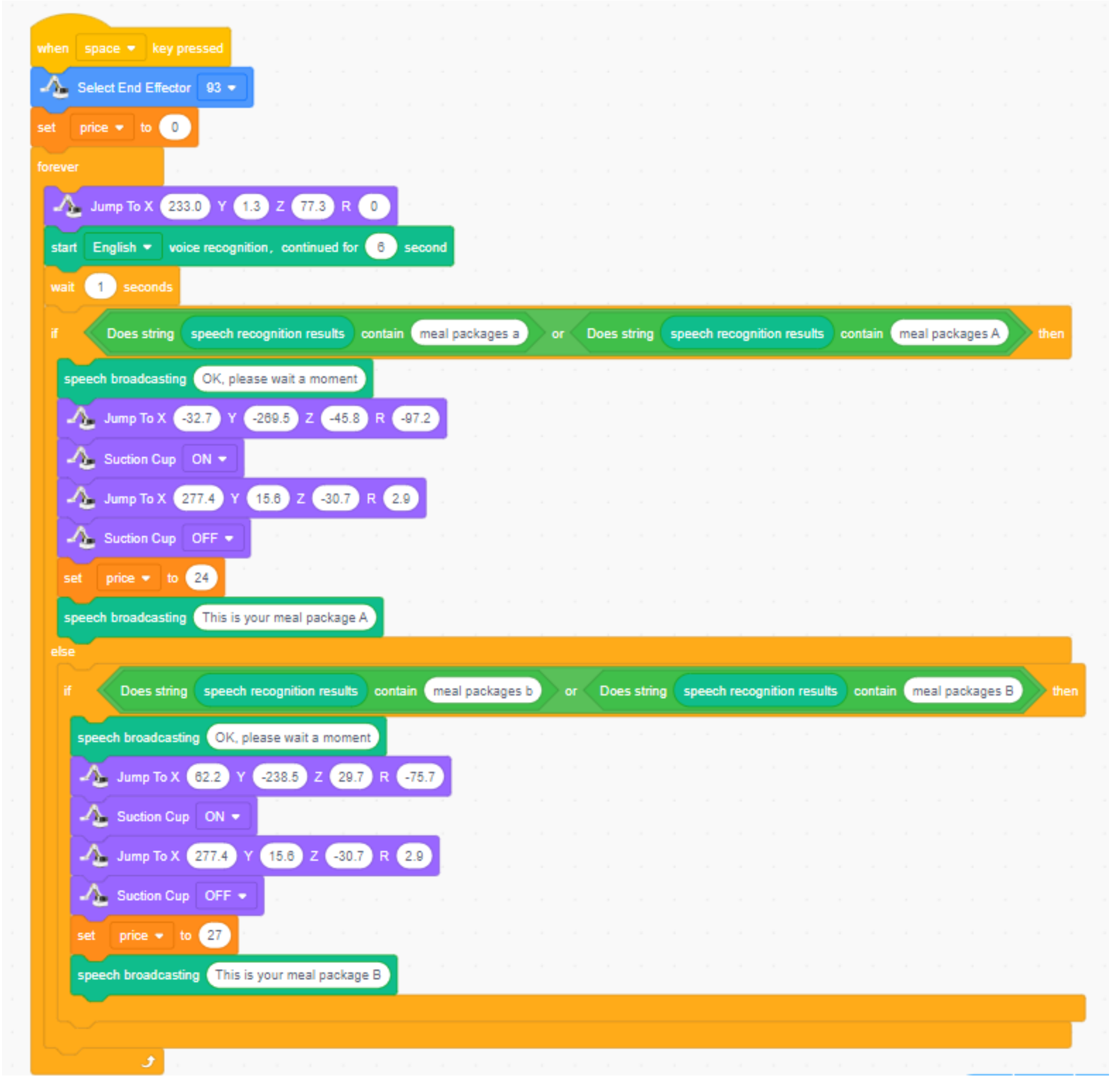


Figure 5.18 Speech recognition food fetching

If a child talks to the robotic arm in Chinese, can the robotic arm understand it? Please think about how to make it happen?

3. Summary

____
____
____
____
(Summarize what you learned in this experiment)

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I've completed the task of creating the variable "price"	
I've completed invoking the "voice recognition" instruction	
I've completed the first judgment structure to determine whether the speech recognition is “meal package A"	
I've completed the task of programming to get the meal package A by the robotic arm	
I've completed the second judgment structure to determine whether the speech recognition is “meal package B"	
I've completed the task of programming to get the set meal B by the robotic arm	

Task 4: Face-Scanning Payment

After ordering meals, you need to pay the fee. This task simulates the application of facial recognition payment. It is assumed that facial data of the students and teachers who eat at the cafeteria has been collected. The task uses John, Kelly, and Mr. Lee as an example. If John ordered meal package A, then the cost of meal package A will be deducted from John's account.

1. Observation

Please observe the operation of the facial recognition payment system and complete the steps of Design Program based on your observation and prompts.

Program design: Initialize > play the voice prompt to face the camera > collect facial data and performing facial recognition > If it is John > deduct the amount from John account ______________
____
____
____
____
(Follow the prompts to complete the remaining steps)

2. Steps

(1) Prepare Hardware

Step 1: Power on the robotic arm, connect the robotic arm to the personal computer (PC), install the end of the suction cup, and install the camera at the end, as shown in Figure 5.19.

Figure 5.19 Installing the camera

Step 2: The position of the robotic arm is the same as that in the task 3.

(2) Design Program

Step 1: Analyze the implementation process of facial recognition payment and draw a flowchart, as shown in Figure 5.20

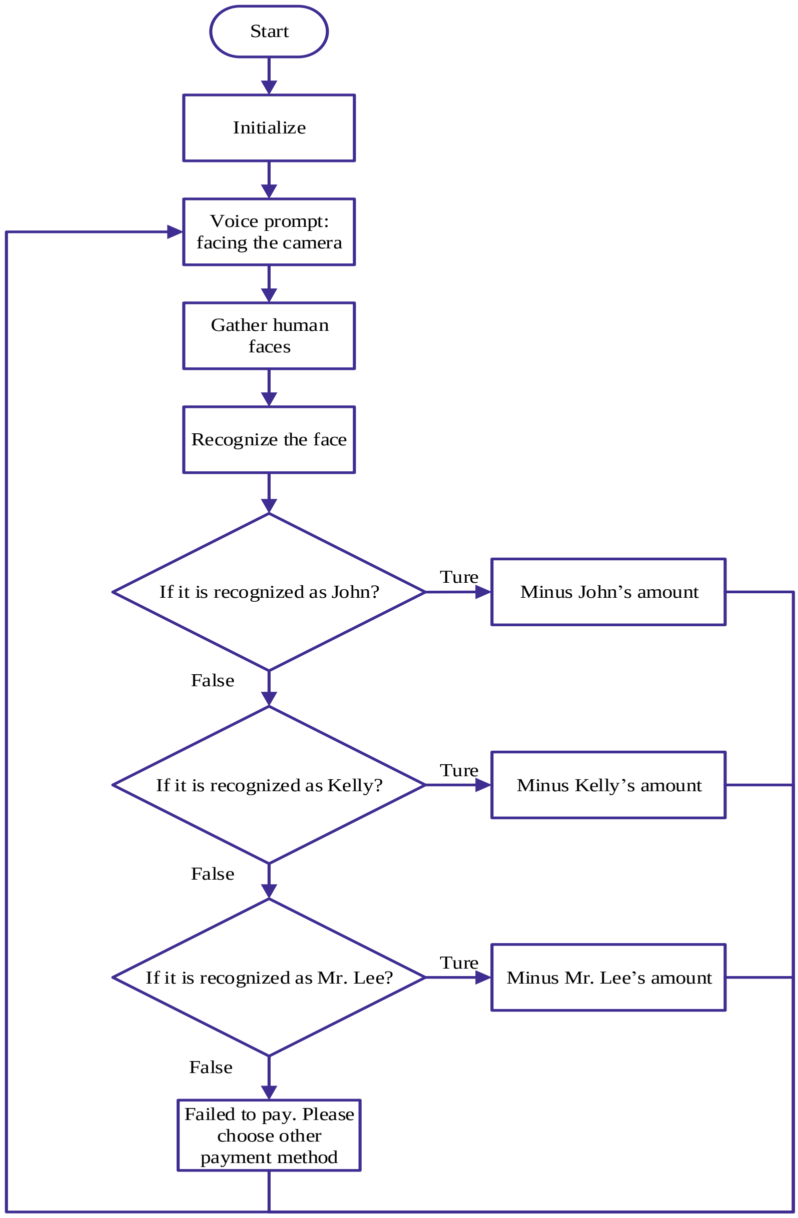


Figure 5.20 flowchart

Step 2: Open the "Voice order" .sb file, collect the facial data and train.

Review task 2 to collect facial data and training.

Step 3: Create three variables, which represent John's account deposit, Kelly's account deposit, and Mr. Lee's account deposit, assuming that the deposits are 100 yuan, 100 yuan, and 200 yuan, as shown in Figure 5.21.

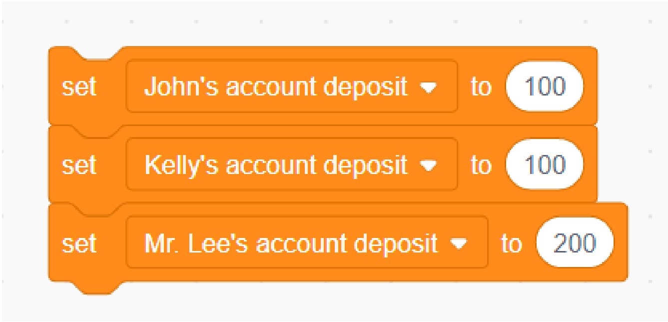


Figure 5.21 Create three variables

Step 4: Set the position of the camera of the robotic arm, and prompt the diner to "please face the camera and complete the facial recognition payment", as shown in Figure 5.22.

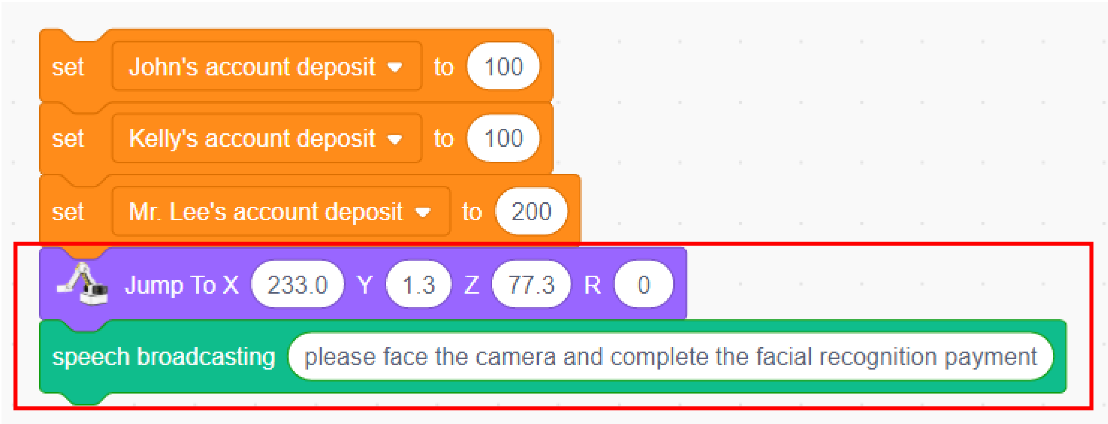


Figure 5.22 Position for photo taking

Step 5: Collect the facial image, and select “Timeout 3s to take picture” on the AI tab page, as shown in Figure 5.23.

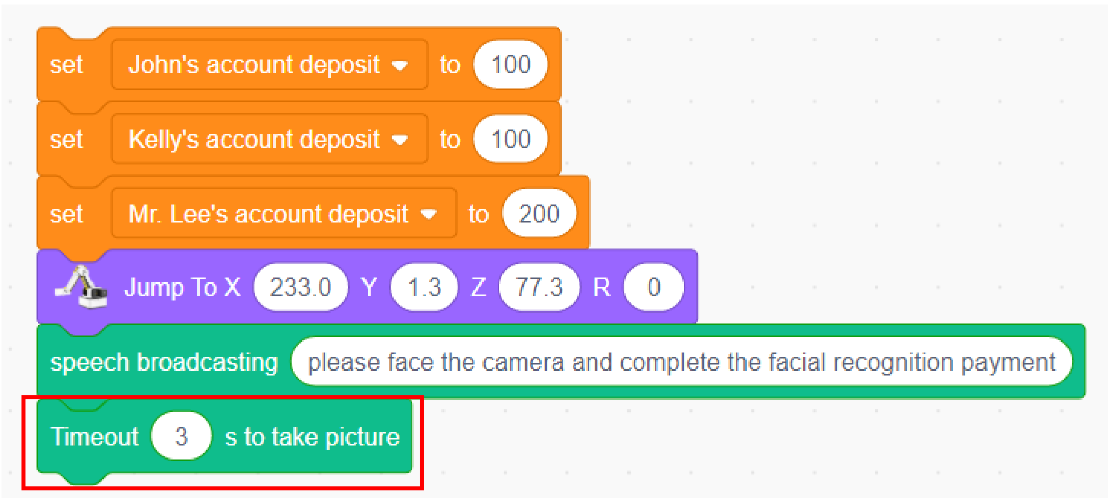


Figure 5.23 Collecting facial images

Step 6: Identify the identity information corresponding to the facial image. If the facial image is John, deduct the amount from John’s account. On the AI tab page, select the "The class name of picture..." block, and embed the "picture" block in this block. The "picture" here is the facial image collected by the camera. In addition, the "join ... and ..." block is in the calculations tab. After dragging out the corresponding blocks, put them together according to a certain logic, as shown in Figure 5.24, where the deduction amount is the price of the meal package, and the remaining amount of John is "John’s deposit minus the price of the set meal".

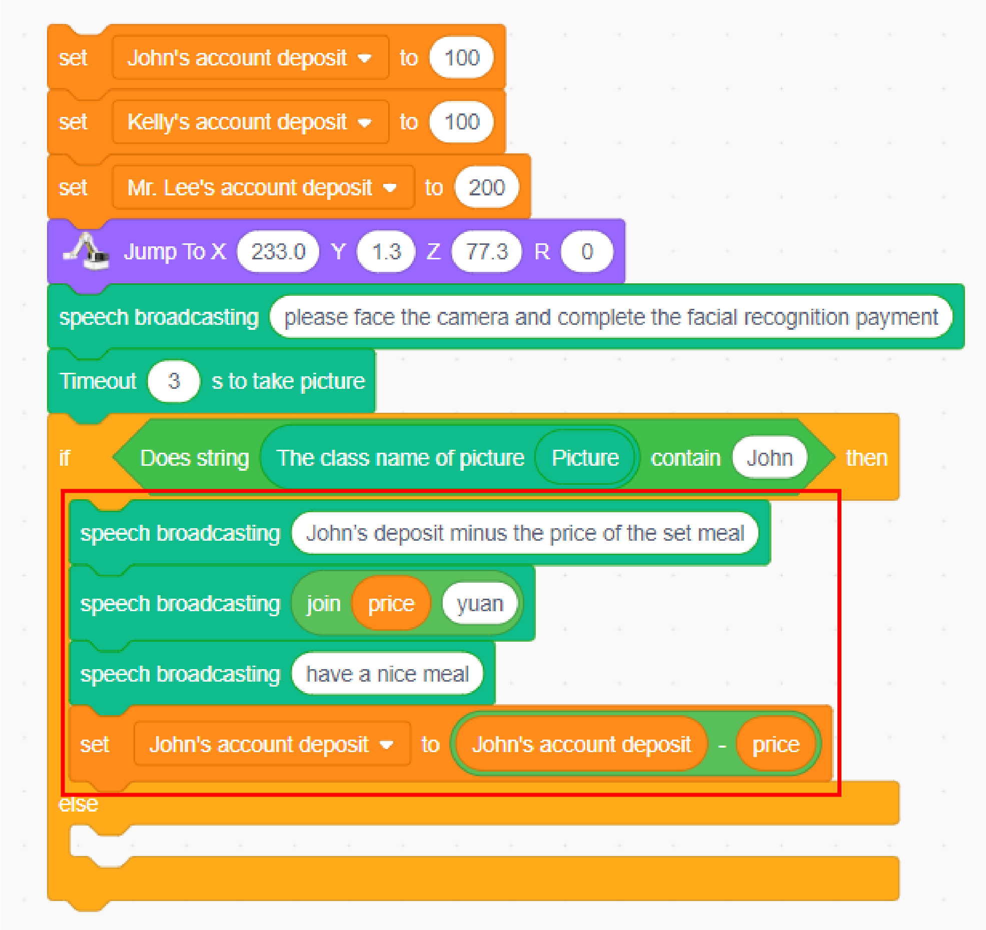


Figure 5.24 Identify facial information and deduct the amount

Step 7: Determine whether the face is Kelly and whether the face is Mr. Lee using the same logic. The program integration is shown in Figure 5.25. "price" is price of the meal package. Once the amount is deducted, each person's deposit will be updated in real time on the stage.

Figure 5.25 Facial recognition payment system

How to put voice ordering together with facial recognition payment to form a "cafeteria assistant" system? Try it by yourself!

3. Summary

(1) Use the "Automatic photo taking after the countdown of ... seconds" block to collect facial images.

(2) Use the "Facial recognition of ..." block to obtain the name corresponding to the face.

____

____

____

(Summarize what you learned in this experiment)

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I’ve completed the task of creating three variables	
I’ve completed the task of collecting facial images	
I’ve completed the task of updating John’s deposit using variables	
I’ve completed the task of updating Kelly’s deposit using variables	
I’ve completed the task of updating Mr. Lee’s deposit using variables	
I’ve written the cafeteria assistant system program	

Experiment 6

Smart Library

Team Name: Team member: Date:

Overview

With the development of artificial intelligence (AI) technologies, many school libraries now use unmanned management systems. Reader identification, borrowing and returning of books can be done automatically on a machine. This experiment simulates an unmanned management system of a smart library. Students can authenticate their identities through facial recognition technology, and complete the process of borrowing and returning books through speech recognition and OCR text recognition.

Objectives

1. By building the smart library system, experience the application of facial recognition technology.
2. By writing the program of the smart library system, master the usage of the facial recognition module in DobotScratch.
3. By building the smart library system, experience the application of speech recognition technology.
4. By writing the program of the smart library system, master the usage of the speech recognition module in DobotScratch.
5. By building the smart library system, experience the application of OCR text recognition technology
6. By writing the program of the smart library system, master the usage of the text recognition module in DobotScratch

Equipment

Icon	Name	Quantity (in units)

	Dobot Magician Lite robotic arm	1
	USB type-C cable	1
	Suction cup	1
	Power adapter	1
	Camera	1
	Book name card	Several

Experiment Requirements

- Take care when using electricity.
- Before the experiment, check whether the experiment equipment is complete and intact. If there is any omission or damage, please report to the teacher.
- Any specific operation in the experiment should be carried out according to the experiment manual. If you have any questions, please promptly ask the teacher.
- In the experiment, the joint of the robotic arm is in a working state after being powered on. Do not force it to move without pressing the unlock button

- Equipment failure during the experiment must be reported to the instructor in time. Do not handle it by yourself.
- After the experiment, the equipment should be arranged well. You are allowed to leave the laboratory only after the inspection by the group leader.

Task 1: Make a Face Recognition System

In conventional libraries, students need to confirm their identities through a library card before they can borrow books. If a student forgets to bring a library card or loses it, he or she cannot borrow any books. Using facial recognition instead of a library card to verify the identity can greatly improve the efficiency in borrowing books and bring great convenience to students. Now, let's prepare the facial recognition system together.

1. Observations

Please observe how the teacher establishes a facial recognition system, and fill in what you see in the blank below.

To establish a facial recognition system, the first step is:____________

The second step is:______________________________________

The third step is:______________________________________

The fourth step is:______________________________________

2. Steps

(1) Prepare Hardware

Step 1: Install a camera at the end of Magician Lite.

Step 2: Power on Magician Lite.

Step 3: Turn on the personal computer (PC).

Step 4: Connect Magician Lite and PC with a USB cable.

Step 5: Connect the camera and PC with a USB cable.

(2) Design Program

Step 1: Design the program flowchart, as shown in Figure 5.1.

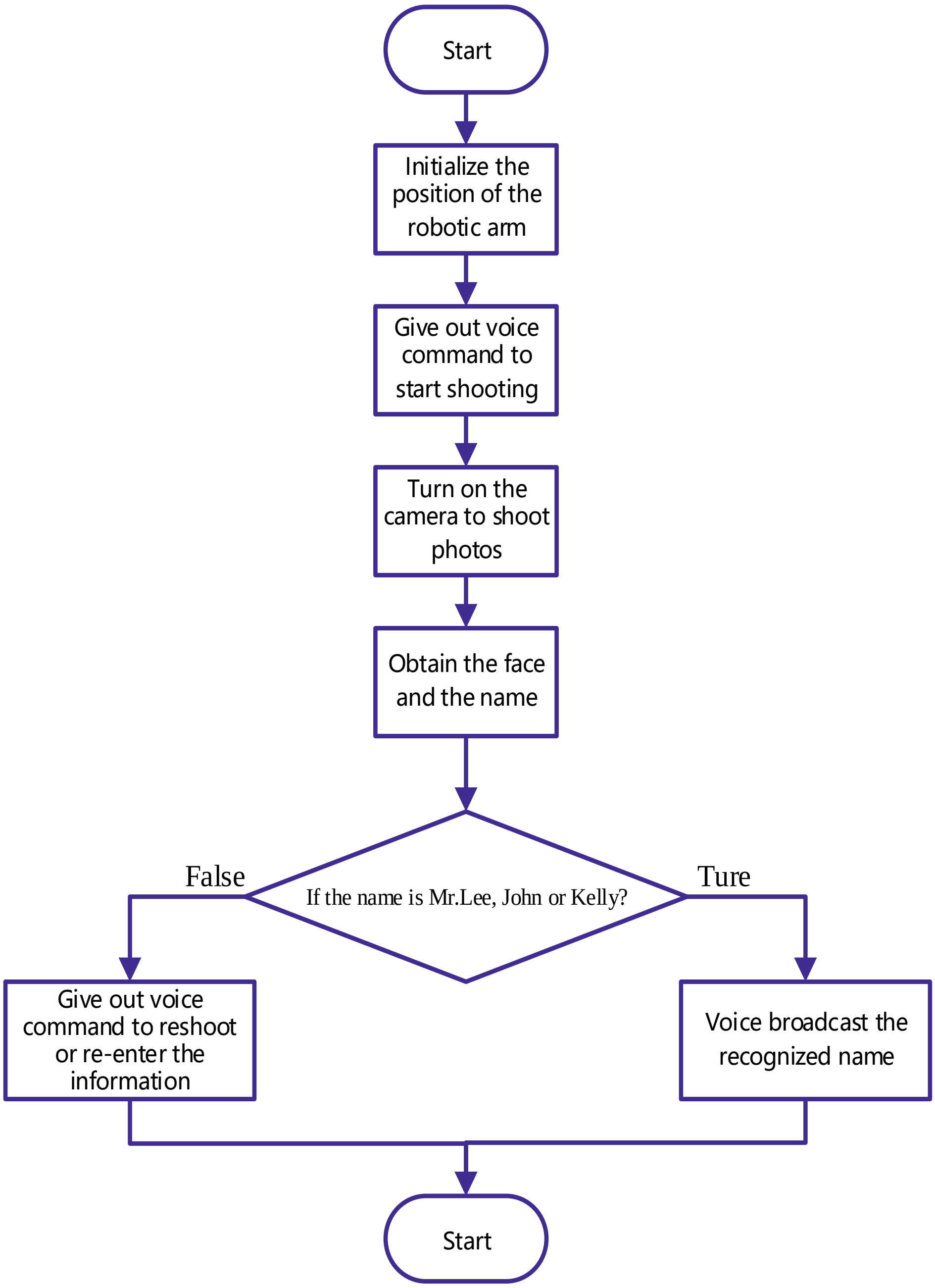


Figure 5.1 Flowchart of identity authentication procedure

Step 2: Open the DobotScratch software.

Step 3: Select Magician Lite and connect it to the corresponding com port.

Step 4: Add the AI module.

Step 5: Click the AI tab, select the facial recognition module, and click "**Edit face data**", as shown in Figure 5.2.

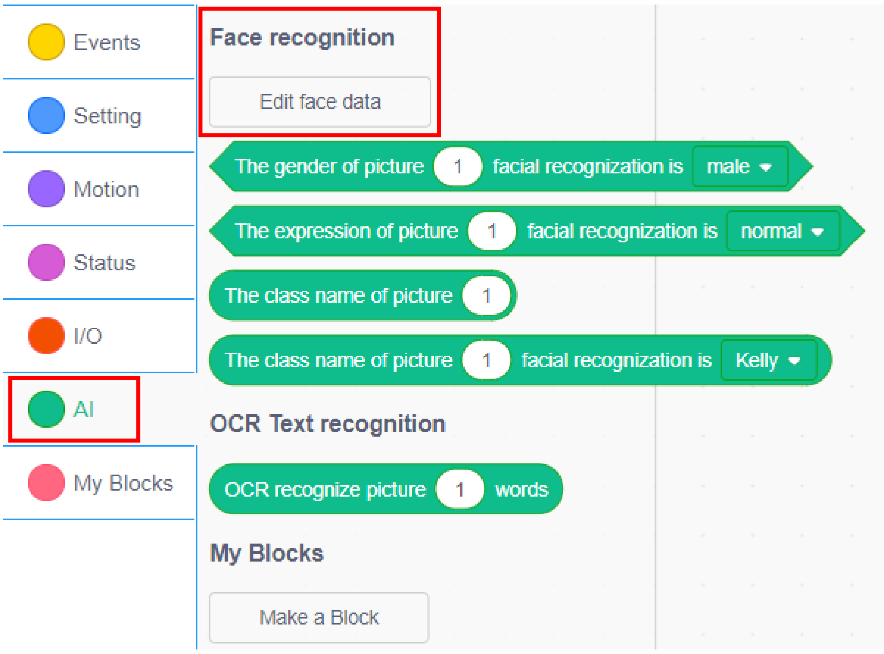


Figure 5.2 face recognition module

Step 6: Create the data set. In this experiment case, three data set labels were established, namely, "Mr. Lee", "John", and "Kelly". Turn on the camera on the robotic arm and add facial images in the corresponding data set, as shown in 5.3.

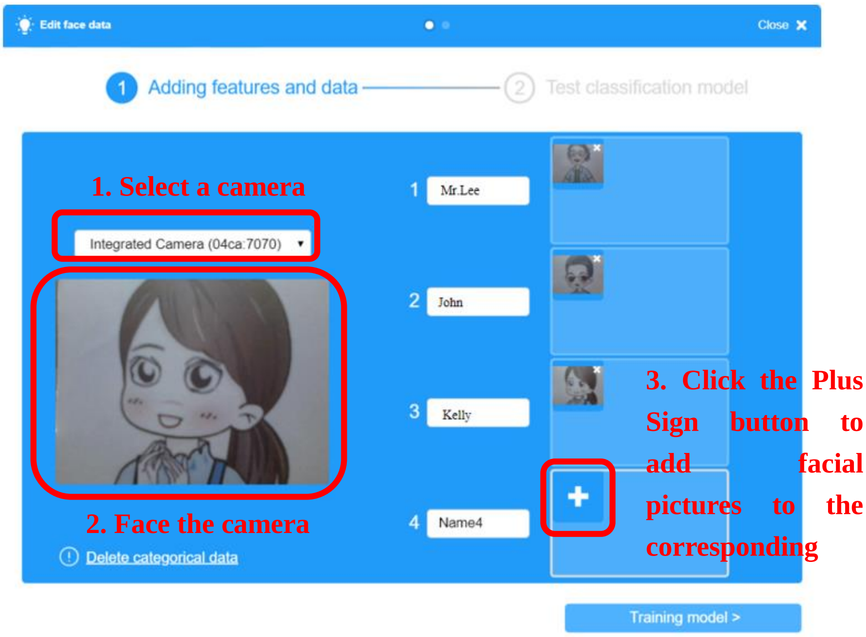


Figure 5.3 Collecting facial data

Step 7: Train the model, as shown in Figure 5.4.

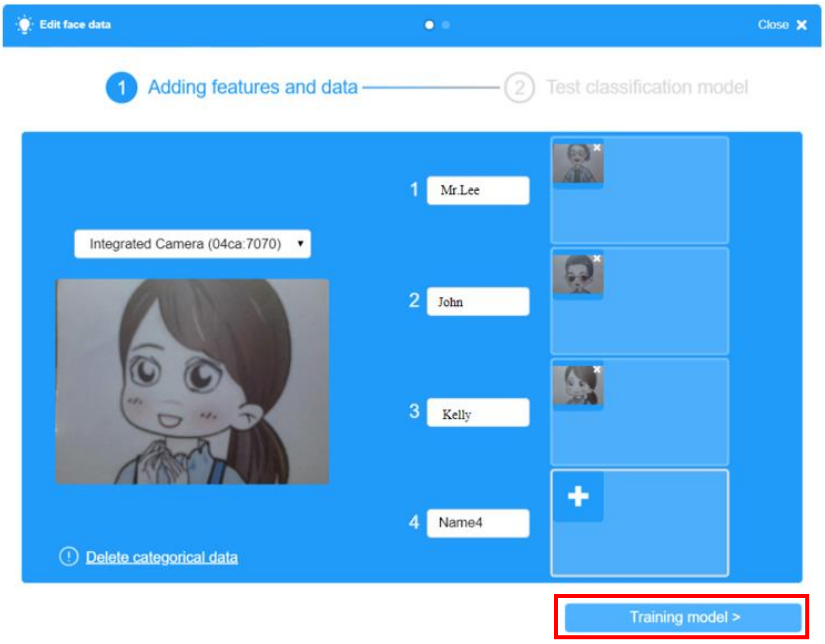


Figure 5.4 Training the facial recognition model

Step 8: Test the facial recognition model. Place the face under the camera and click the **test** button. The right side of the test window shows the computer-predicted probability that the face matches each name. The likelier the name is to be correct, the more accurate the facial recognition model, as shown in Figure 5.5.

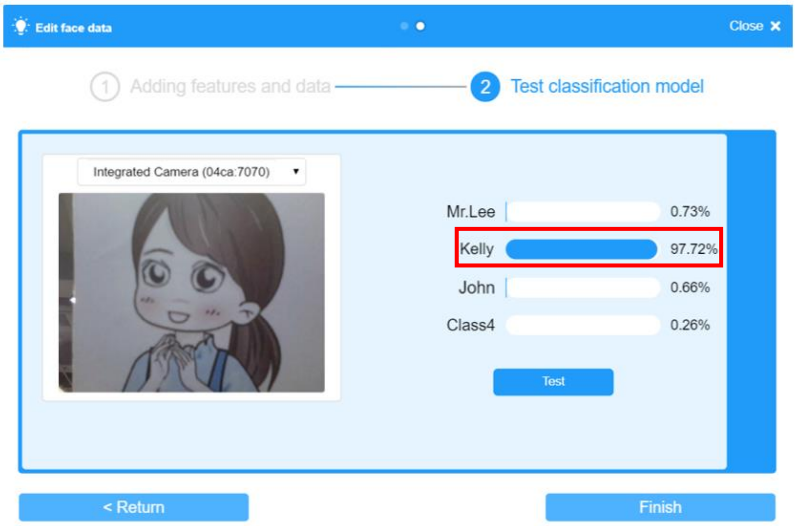


Figure 5.5 Testing the facial recognition model

Step 9: Write an identity authentication program. Move the robotic arm to a suitable position, and turn on the camera to take a picture of the face, as shown in Figure 5.6.

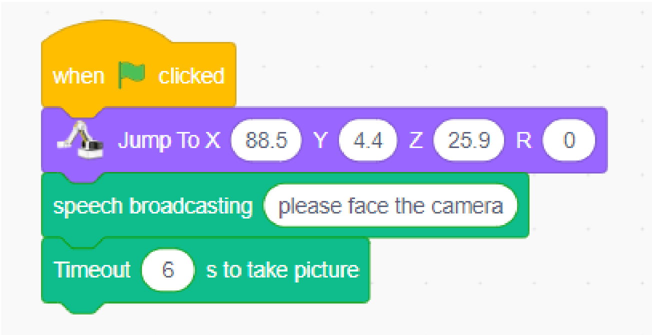


Figure 5.6 Program of taking a photo with a camera

Step 10: Obtain the name of the facial image and determine whether the name is in the dataset, as shown in Figure 5.7.

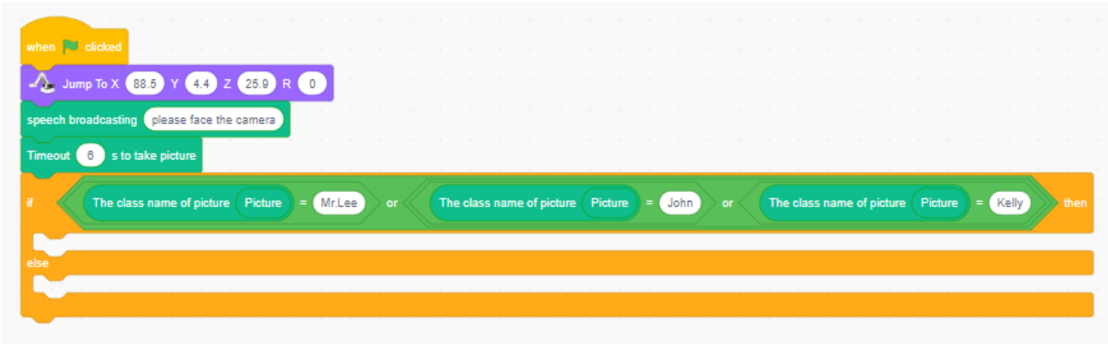


Figure 5.7 Obtaining test pictures with a camera

Step 11: Play the corresponding name if the facial picture is in the dataset; otherwise remind the user to take a new photo or re-enter the information, as shown in Figure 5.8.

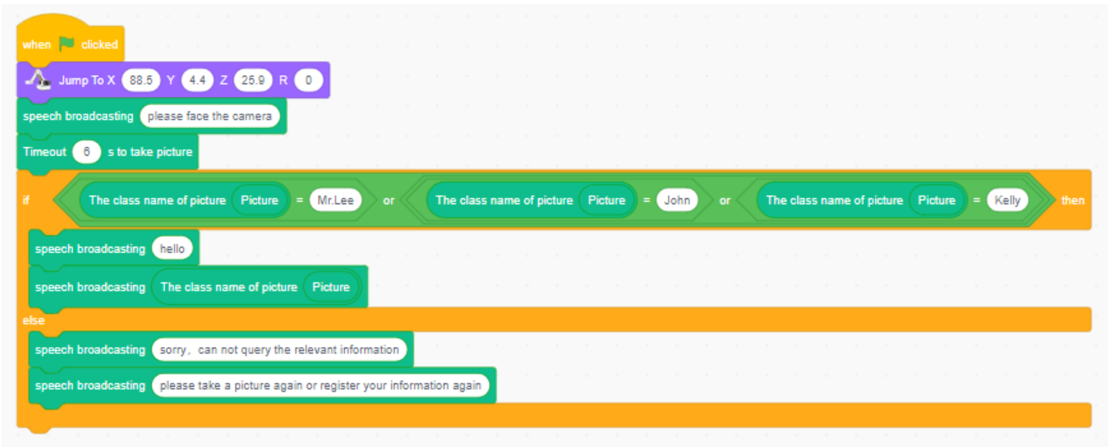


Figure 5.8 Authenticating the identity

3. Conclusions

(5) The steps to establish a facial recognition system are: ________

____

(6) How to build facial data: ______________________________

____

(7) The instructions for the facial recognition are: ______________

____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I've known the steps to achieve facial recognition	
I've collected facial data	
I've trained a facial recognition model	
I've tested the facial recognition model	
I've completed the identity authentication task with facial recognition	

Task 2: Establish Voice-Based Book Borrowing System

When students go to the library to borrow books, they need to find the location of the books by themselves, so the efficiency is not high. In order to improve the efficiency in borrowing books, you can help pick up the books by voice-controlled robots. Now, let's establish a voice-based book borrowing system together.

1. Observations

Please observe how the teacher makes speech recognition work, and analyze the steps of establishing a speech recognition system.

2. Steps

(1) Prepare Hardware

Step 1: Design the experiment map and place the equipment. The reference of this experimental map is shown in Figure 5.9.

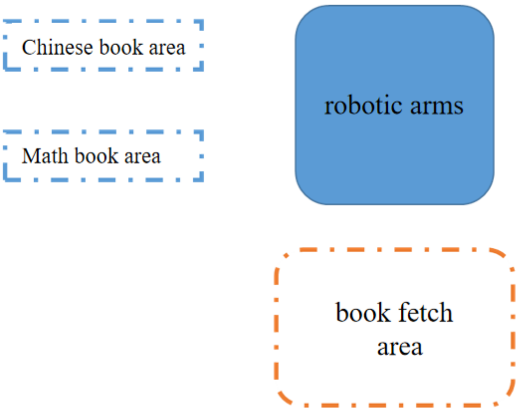


Figure 5.9 Map of the smart library

Step 2: Power on Magician Lite.

Step 3: Turn on the personal computer (PC).

Step 4: Connect Magician Lite and PC with a USB cable.

(2) Design Program

Step 1: Design the program flowchart, as shown in Figure 5.10.

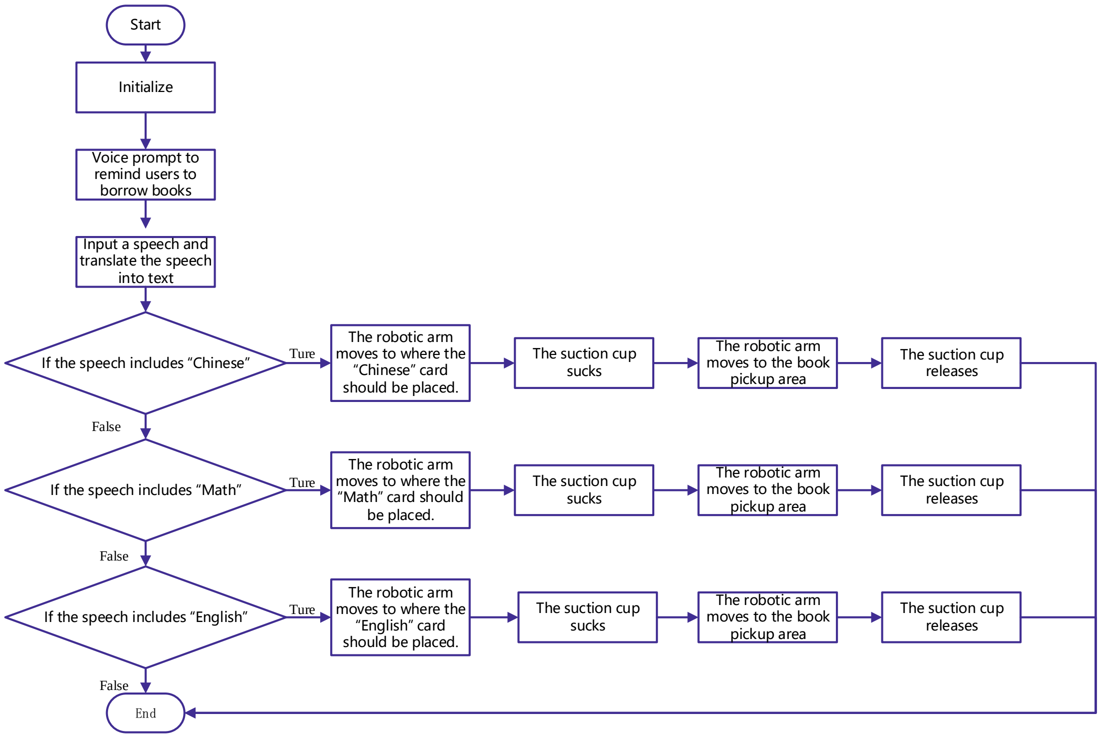


Figure 5.10 Flowchart of the voice-based book borrowing program

Step 2: Open the DobotScratch software.

Step 3: Select Magician Lite and connect it to the corresponding com port.

Step 4: Add the AI module.

Step 5: Check whether the speech recognition module works. Select the speech recognition module in the AI module, click the O**pen speech recognition** button, and then click **Start**, as shown in Figure 5.11.

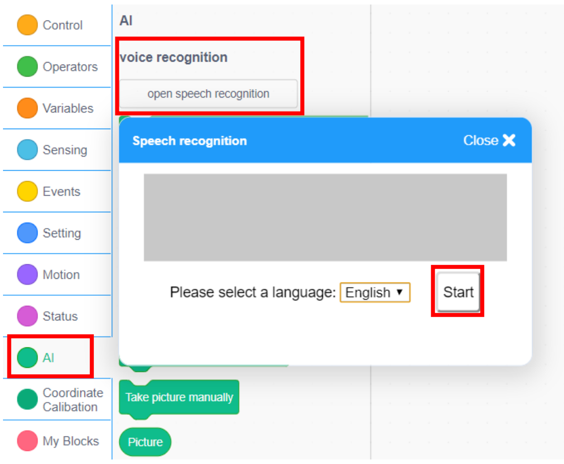


Figure 5.11 Recording the voice

Step 6: After recording the voice, click **Stop** to check whether the text of the recognition result matches the voice you've recorded. If yes, the speech recognition module can be used normally, as shown in Figure 5.12.

Figure 5.12 Testing the voice module

Step 7: Start writing the book borrowing program. Set the initial position of the robotic arm, and prompt the student by voice to enter the title of the book to be borrowed, as shown in Figure 5.13.

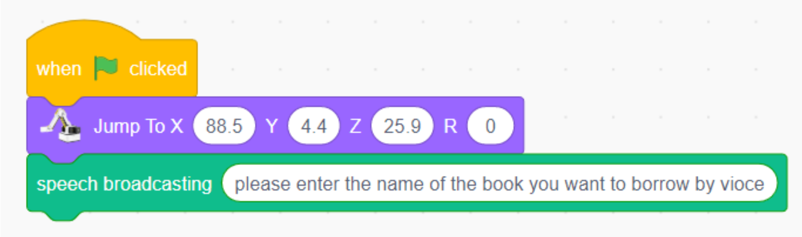


Figure 5.13 Initializing the position of the robotic arm

Step 8: Record the voice and recognize the voice information to convert it into text. As shown in Figure 5.14.

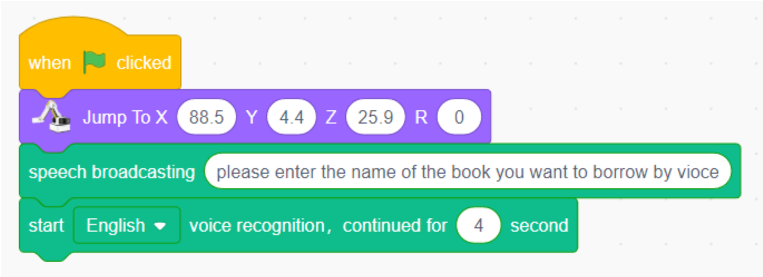


Figure 5.14 Recognizing student's voice.

Step 9: Determine whether the student's voice contains the word "Chinese". If yes, move the robotic arm to the "Chinese" book area and grab the book to move it to the book fetch area. The locations of the Chinese book area and the book fetch area can be obtained by moving the robotic arm in advance, as shown in Figure 5.15.

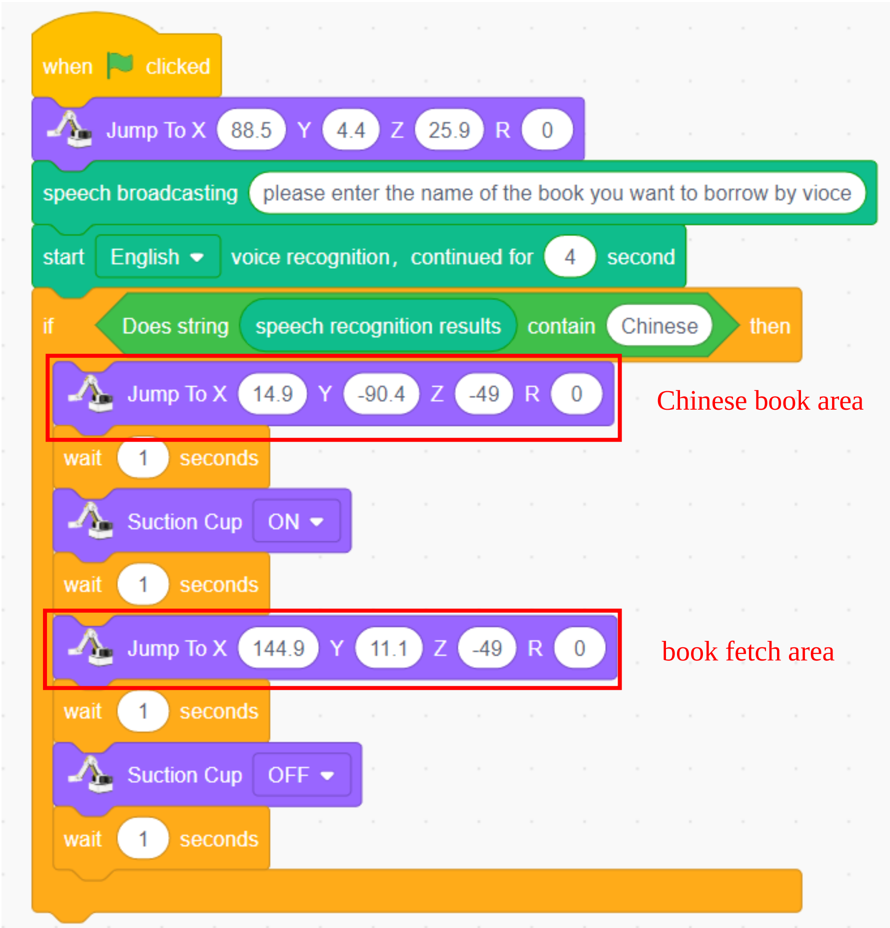


Figure 5.15 Borrowing the Chinese book

Step 10: Determine whether the student's voice contains the word "math". If yes, move the robotic arm to the mathematics book area and grab it to move it to the book fetch area, as shown in Figure 5.16.

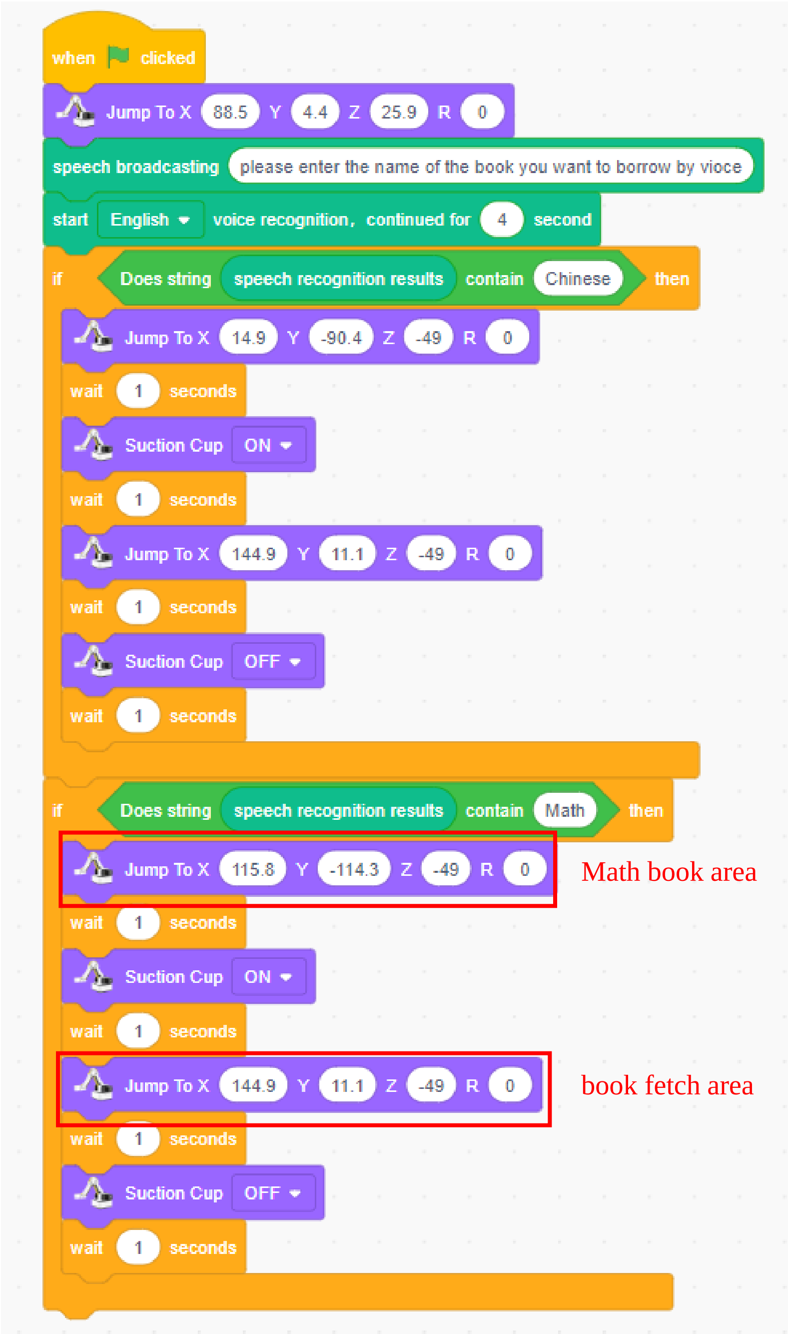


Figure 5.16 Borrowing the mathematics book

Step 11: Determine whether the student's voice contains the word "English". If yes, move the robotic arm to the English book area and grab the book to move it to the book fetch area, as shown in Figure 5.17.

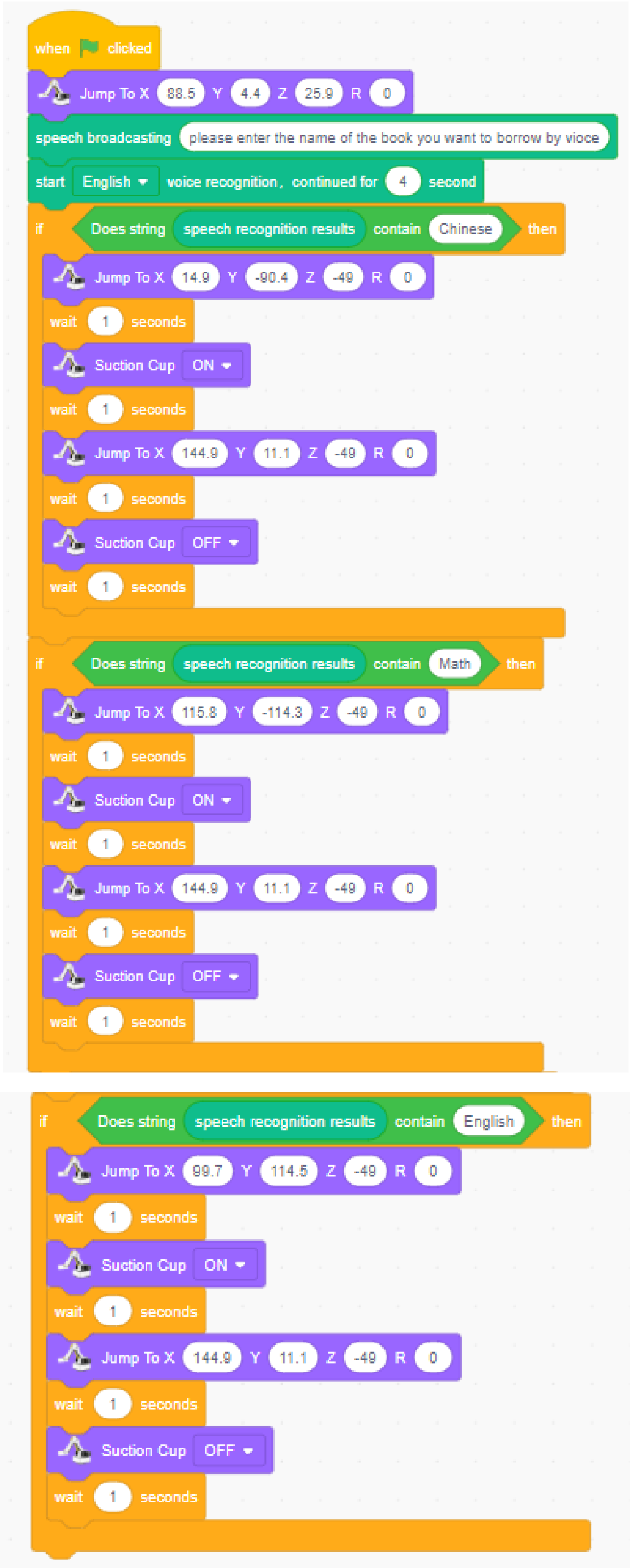


Figure 5.17 Borrowing the English book

3. Conclusions

(6) The steps for speech recognition are: ______________________
____

(7) The instructions for speech recognition are: ________________
____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I’ve known the steps for speech recognition	
I've completed the test for speech recognition	
I've completed the task of borrowing books by voice	

Task 3: Establish OCR Book Returning System

In the previous task, we used a voice-controlled robot to take books, which makes it faster to borrow books. Can the robot return books as well? The answer is yes. We can use OCR text recognition technology to let the robot recognize the book title, and then place it in the corresponding area. Now, let's implement the process of returning books by a robot together.

1. Observations

Please observe how the teacher performs text recognition using the OCR module, and then fill the key steps in the blank below.

To make OCR text recognition work, the first step is:____________

The second step is:______________________________________

2. Steps

(1) Prepare Hardware

The Prepare Hardware for Task 3 is the same as that for Task 2. Please refer to for Task 2.

(2) Design Program

Step 1: Design the program flowchart, as shown in Figure 5.18.

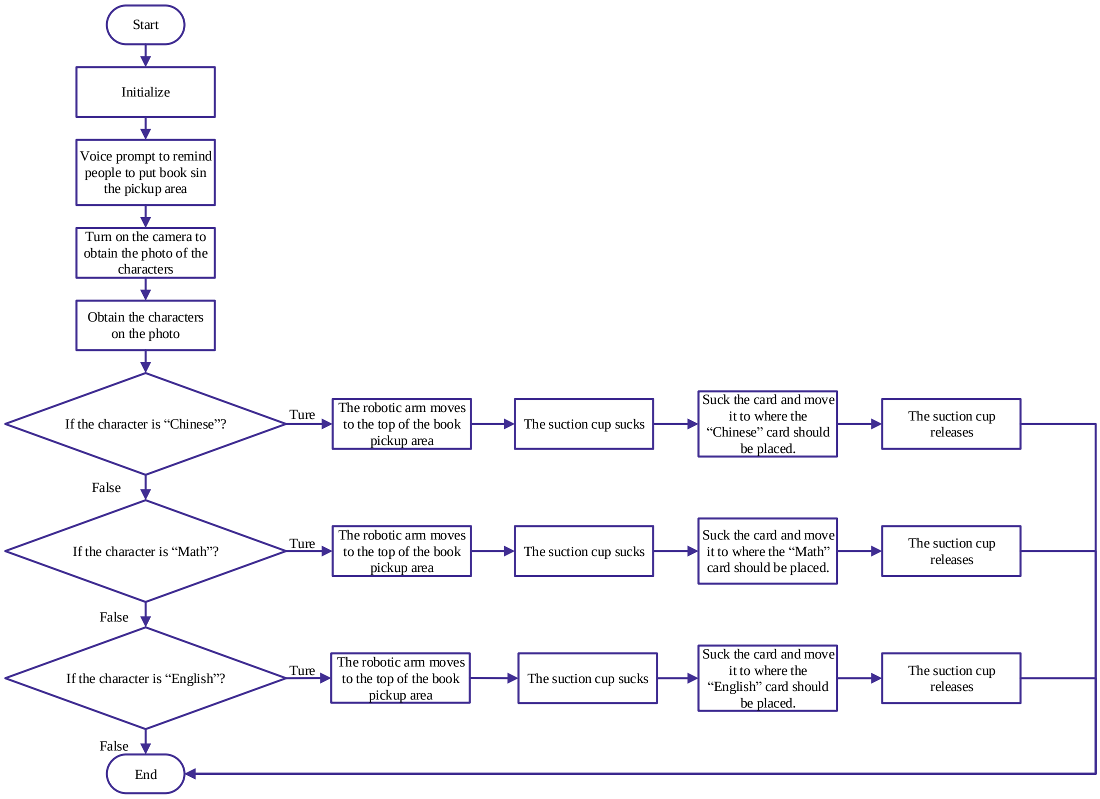


Figure 5.18 The flowchart of OCR book returning program

Step 2: Open the DobotScratch software.

Step 3: Select Magician Lite and connect it to the corresponding com port.

Step 4: Add the AI module.

Step 5: Start writing the OCR book returning program. Set the initial position of the robotic arm, prompt the student by voice to place the book in the book pickup area, and open the camera, as shown in Figure 5.19.

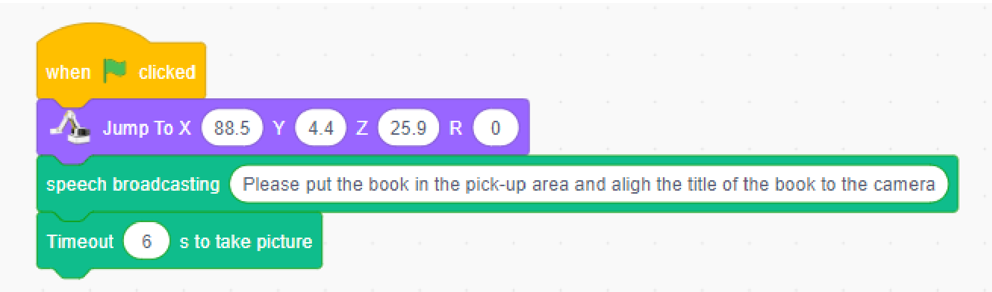


Figure 5.19 Initializing the position of the robotic arm

Step 6: Check the accuracy of OCR recognition. Create a variable "test", click on the "**Variables**" tab, and drag the "Set test to" block to connect to the previous one, as shown in Figure 5.20.

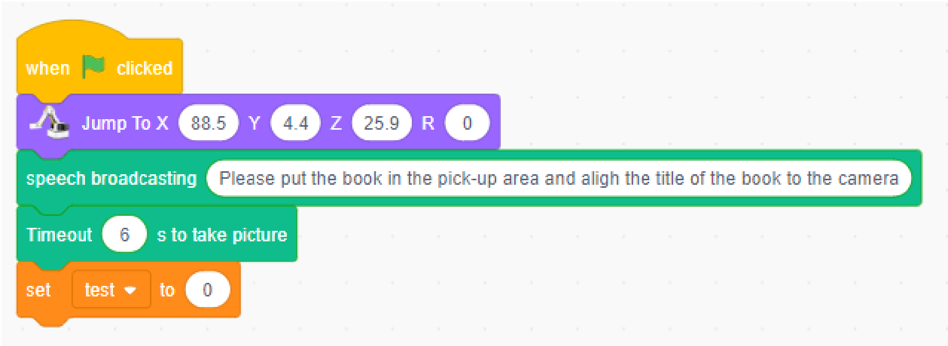


Figure 5.20 Defining the variable

Step 7: Store the text recognized by OCR in the variable *test*. Click on the **AI** tab and drag "OCR-recognize picture...words" block onto the variable *test*; click the **AI** tab, and drag the "Picture" block to the OCR recognition block, as shown in Figure 5.21.

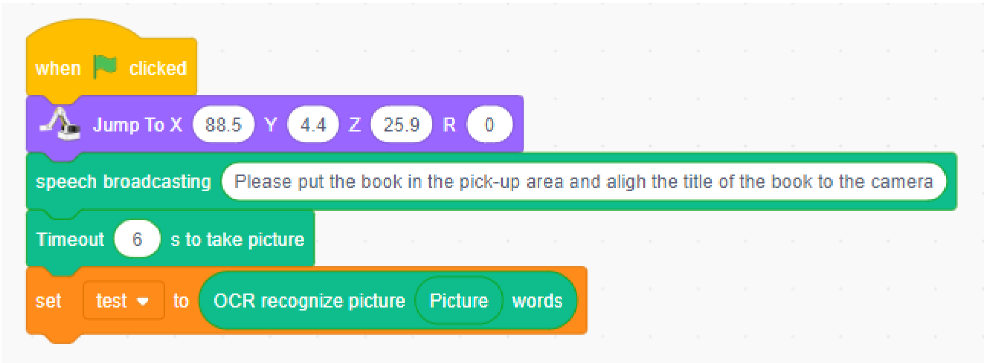


Figure 5.21 Recognizing text in the picture

Step 8: Click the green flag to run the program. Place the book name card under the camera, and check whether the value of the variable *test* on the stage which lies in the upper left corner of the software is equal to the text on the card, as shown in Figure 5.22.

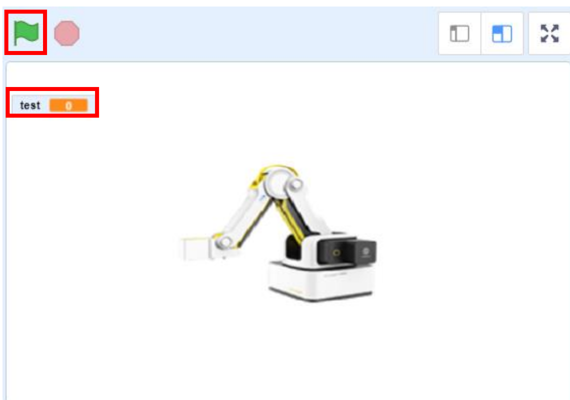


Figure 5.22 Testing the accuracy of OCR recognition

> 👉 You can write some words on a blank card, and then place it under the camera to test whether the robot can recognize the text correctly.

Step 9: Apply the OCR identification module. Determine whether the book name recognized is "Chinese". If yes, suck the Chinese book card to the storage area of the Chinese book, as shown in Figure 5.23.

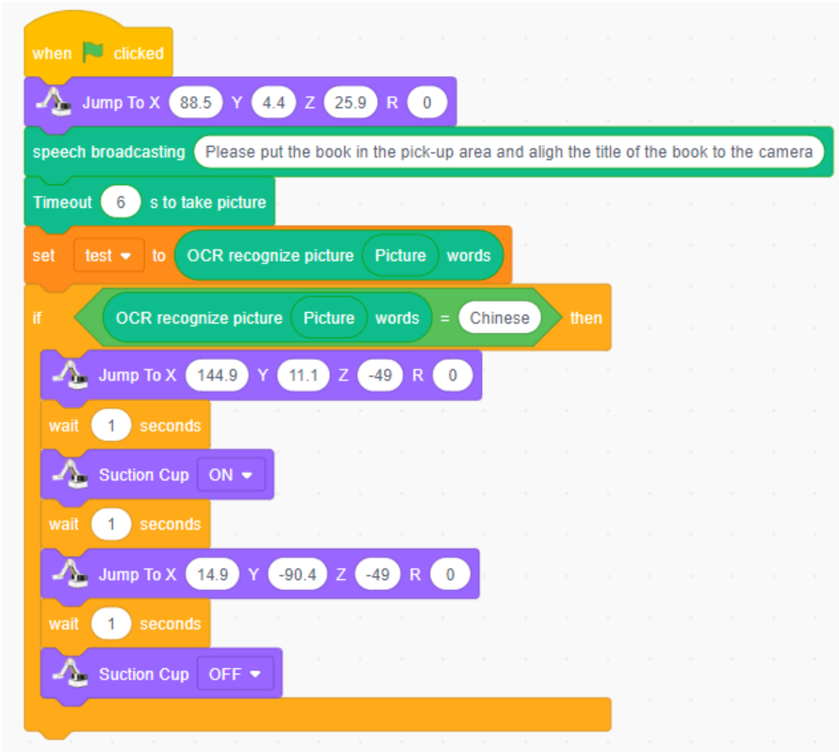


Figure 5.23 Determining whether it is a Chinese book

Step 10: Determine whether the book name recognized is "Mathematics". If yes, suck the mathematics book card to the storage area of the mathematics book, as shown in Figure 5.24.

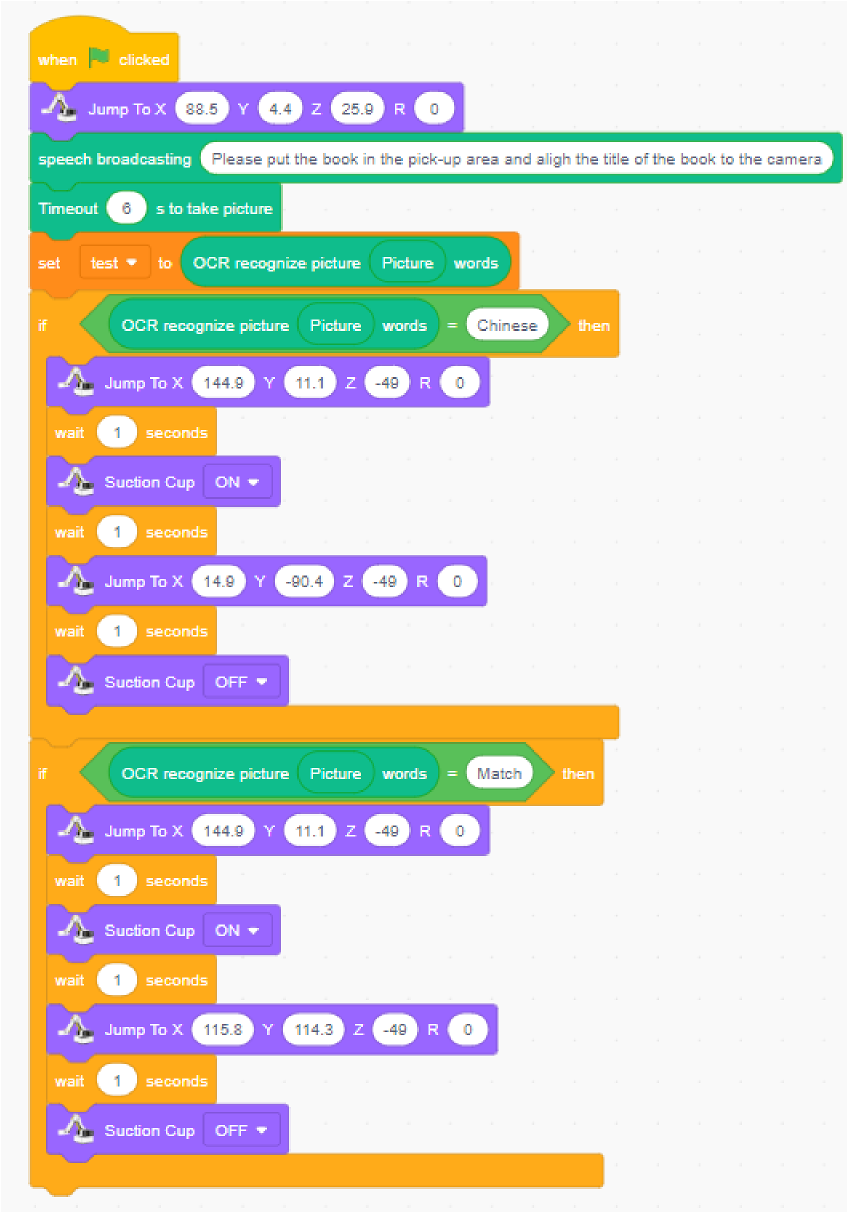


Figure 5.24 Determining whether it is a mathematics book

Step 11: Determine whether the book name recognized is "English". If yes, suck the English card to the storage area of the English book, as shown in Figure 5.25.

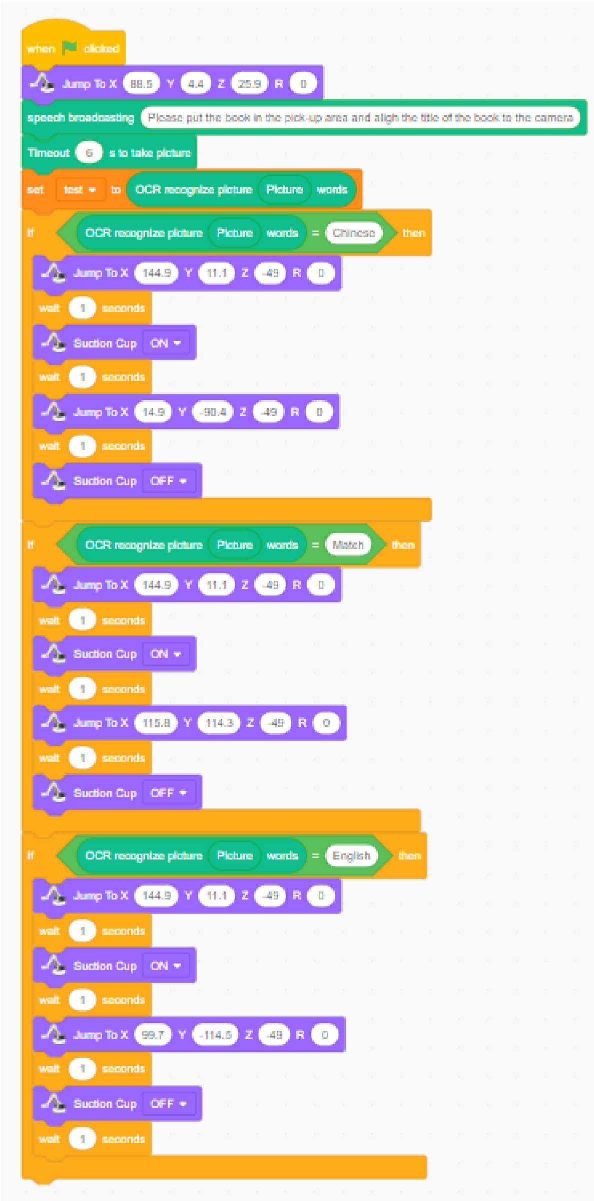


Figure 5.25 Determining whether it is an English book

3. Conclusions

(1) OCR text recognition instructions: ____________________________

____

(2) What are the factors that affect OCR text recognition: __________

____

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment Content	Completion Status
I've known the process of OCR text recognition	
I've tested OCR text recognition	
I've finished the task of returning books by OCR recognition	

Task 4: Establish A Complete Smart Library System

In the previous tasks, we have completed the facial recognition identity authentication system, voice-based book borrowing system and OCR book returning system in the library. Now, let's integrate the tasks finished above and establish a complete smart library system.

1. Observations

Please observe how the teacher sets up the smart library system and analyze the steps to complete the task.

To establish a complete smart library system, the first step is:______

The second step is:____

The third step is:____

2. Steps

(1) Prepare Hardware

The Prepare Hardware for Task 4 is the same as that for Task 2. Please refer to Task 2.

(2) Design Program

Step 1: Design the program flowchart, as shown in Figure 5.26.

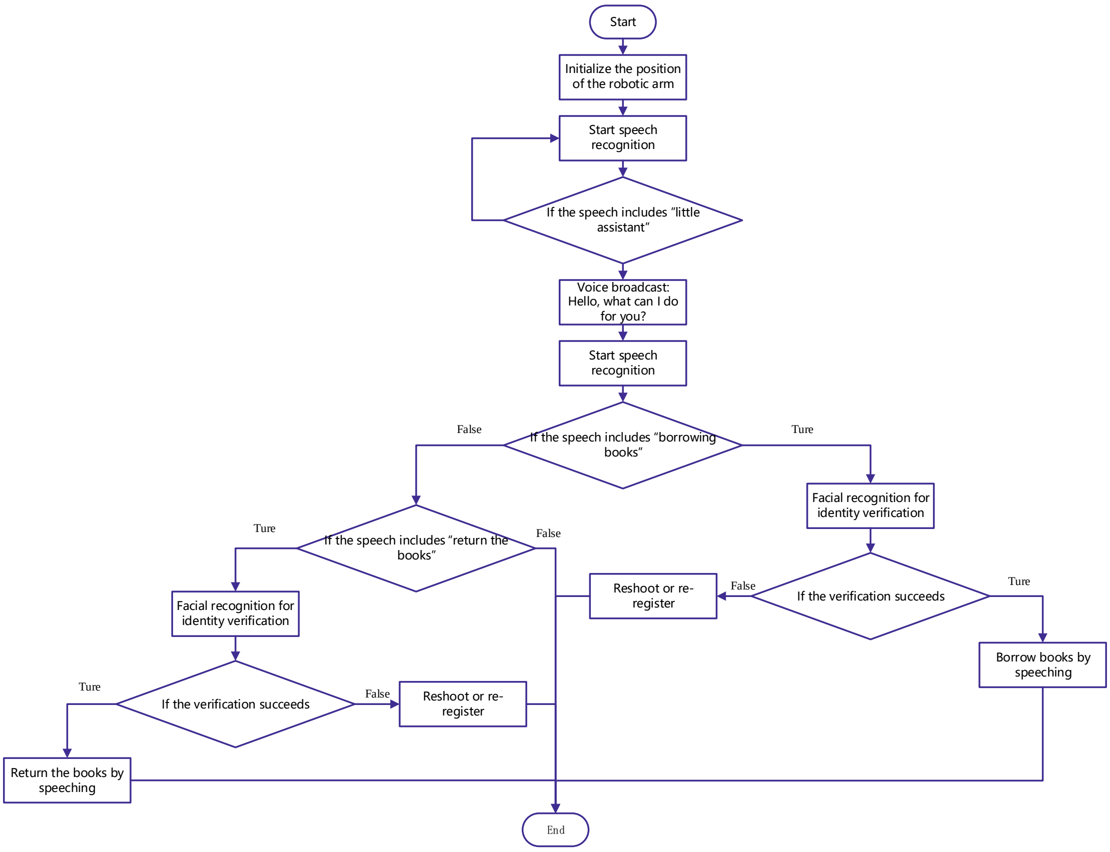


Figure 5.26 Flowchart of the smart library program

Step 2: Write the code. Write a voice borrowing function. Click the My **Blocks** tab, choose Making a New Block, name the block as Voice-based Book Borrowing, and then click **Finish**, as shown in Figure 5.27.

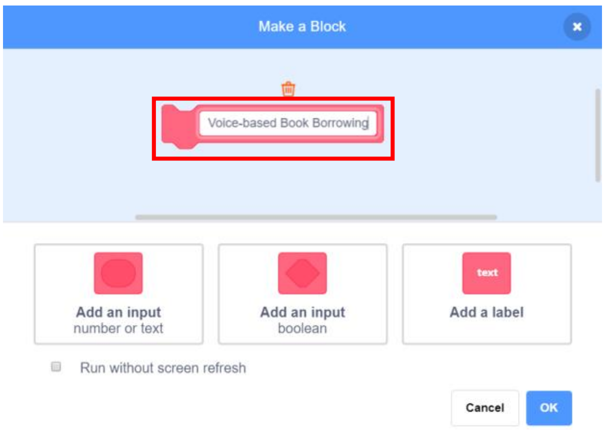


Figure 5.27 Creating the block of book borrowing by voice

Step 3: Write the content of the block of book borrowing by voice. Click the **My Blocks** tab, drag the "Voice-based Book Borrowing" block to the code area, and place the code for book borrowing by voice under the "Voice-based Book Borrowing" block, as shown in Figure 5.28.

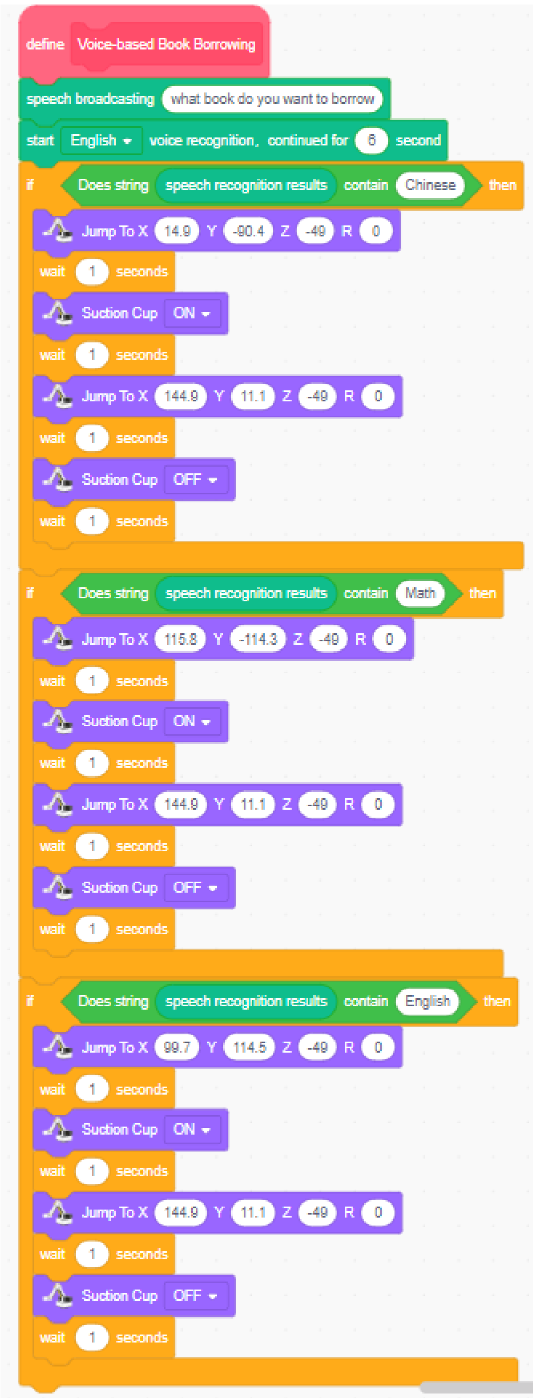


Figure 5.28 Defining the block of voice-based book borrowing

Step 4: Likewise, make an OCR book returning block, as shown in Figure 5.29.

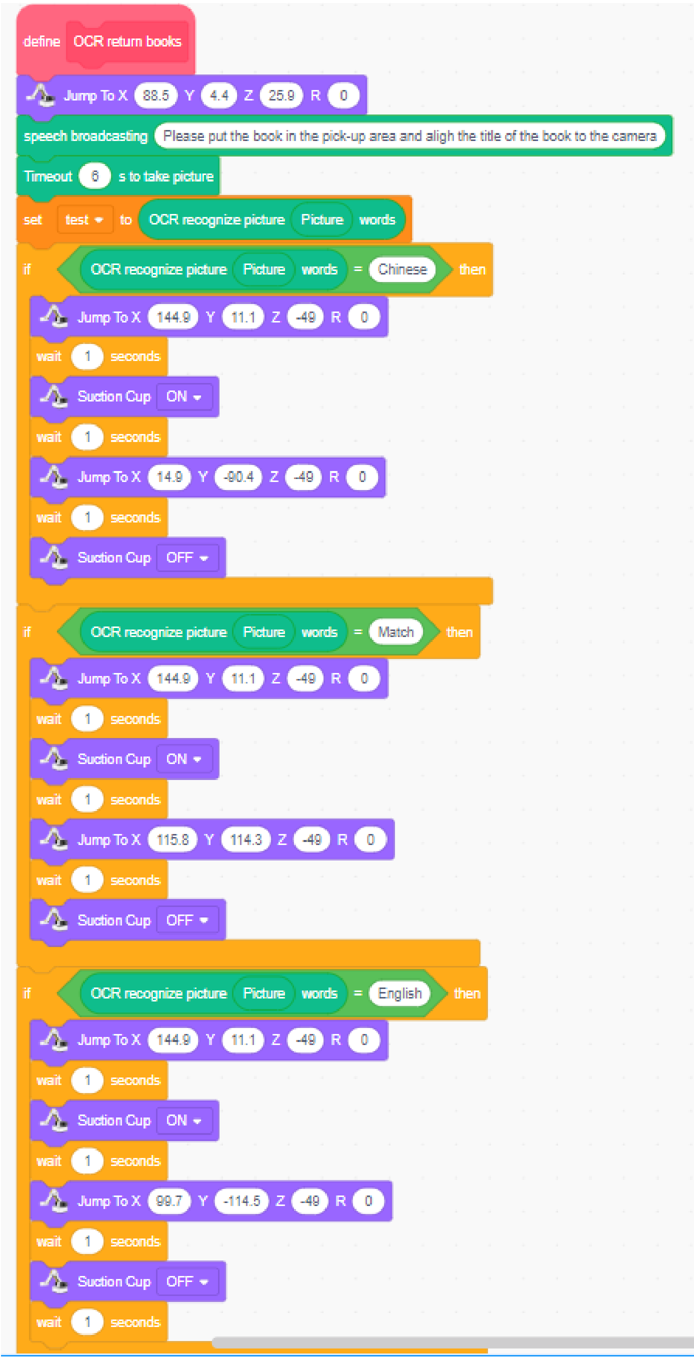


Figure 5.29 Defining the OCR book returning block

Step 5: Initialize the position of the robotic arm, and wake up the smart library system by voice. The program is shown in Figure 5.30.

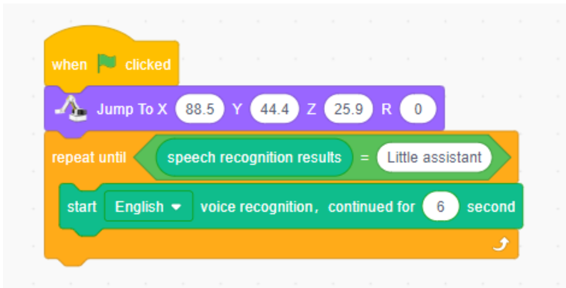


Figure 5.30 Program of waking up by voice

Step 6: Choose book borrowing or book returning service by voice, as shown in Figure 5.31.

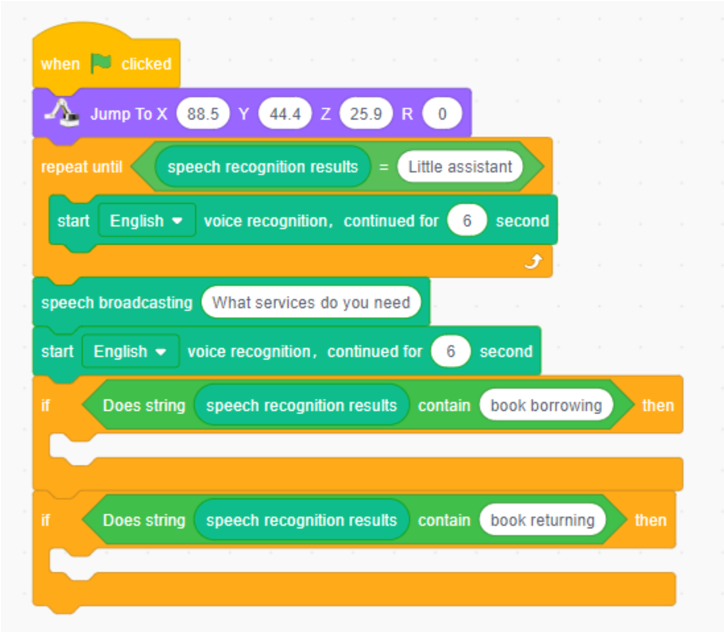


Figure 5.31 Determining whether to borrow or return books

Step 7: Authenticate the identity before borrowing a book. If the authentication succeeds, then perform voice-based borrowing. If the authentication fails, verify or register the information again, as shown in 5.32.

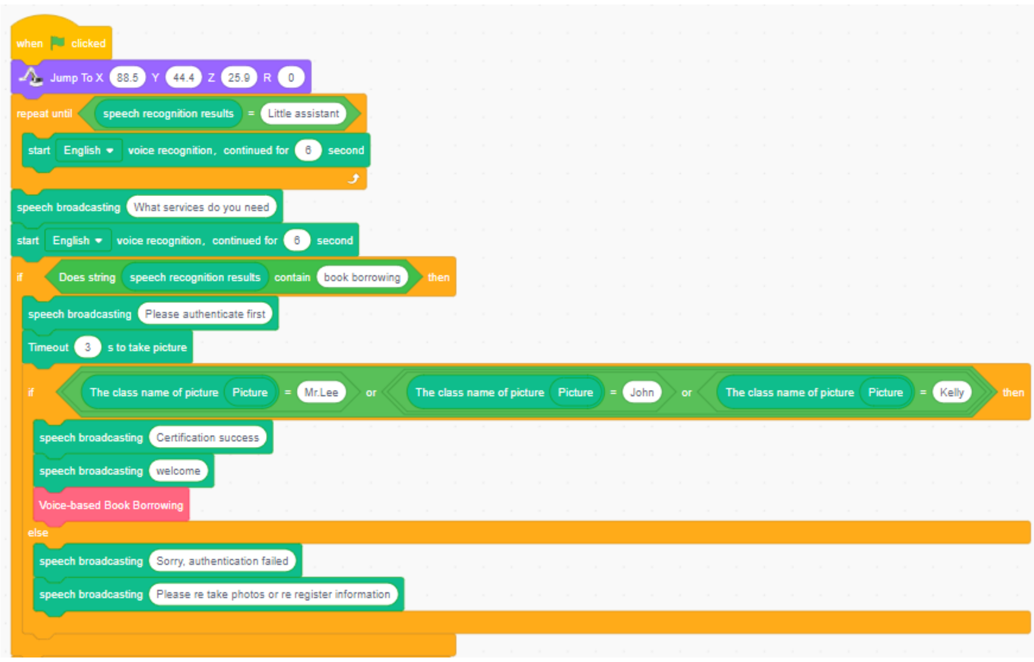


Figure 5.32 Book borrowing program

Step 8: Authenticate the identity before borrowing a book. If the authentication succeeds, then return the book by OCR. If the authentication fails, verify or register the information again, as shown in 5.33.

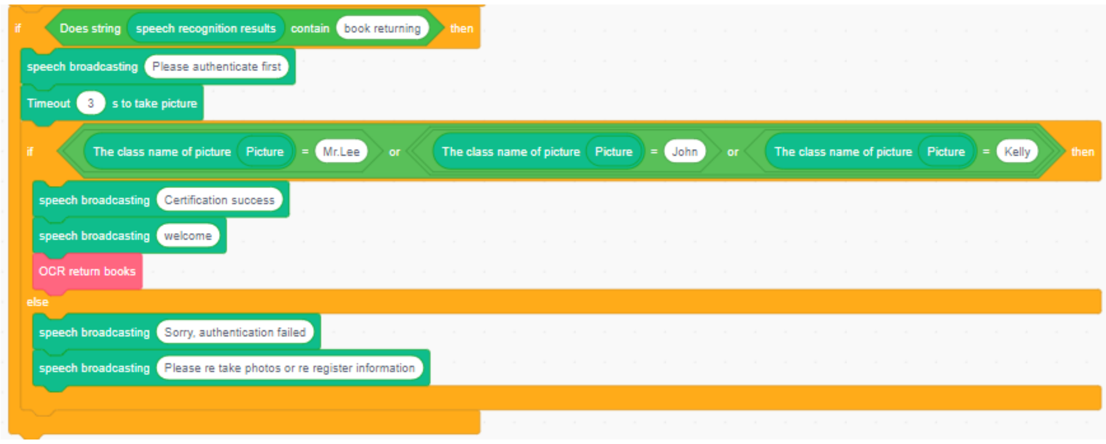


Figure 5.33 Book returning program

3. Conclusions

(1) Steps of establishing a smart library: ______________________

(2) The instructions for self-making blocks are: __________________

4. Self-Assessment

Check the content finished in the experiment tasks, tick (√) the finished ones, and circle (◯) for the unfinished ones.

Assessment content	Completion Status
I've used instructions for customizing a block	
I've implemented a complete smart library system	